ASEC Report 5월

® ASEC Report

2007. 6

안철수연구소의 시큐리티대응센터(AhnLab Security Emergency response Center)는 악성코드 및 보안위협으로부터 고객을 안전하게 지키기 위하여 바이러스 분석 및 보안 전문가들로 구성되어 있는 조직이다.

이 리포트는 ㈜안철수연구소의 ASEC에서 국내 인터넷 보안과 관련하여 보다 다양한 정보를 고객에게 제공하기 위하여 바이러스와 시큐리티의 종합된 정보를 매월 요약하여 리포트 형태로 제공하고 있다.

I. ASEC 월간 통계

(1) 5월 악성코드 통계

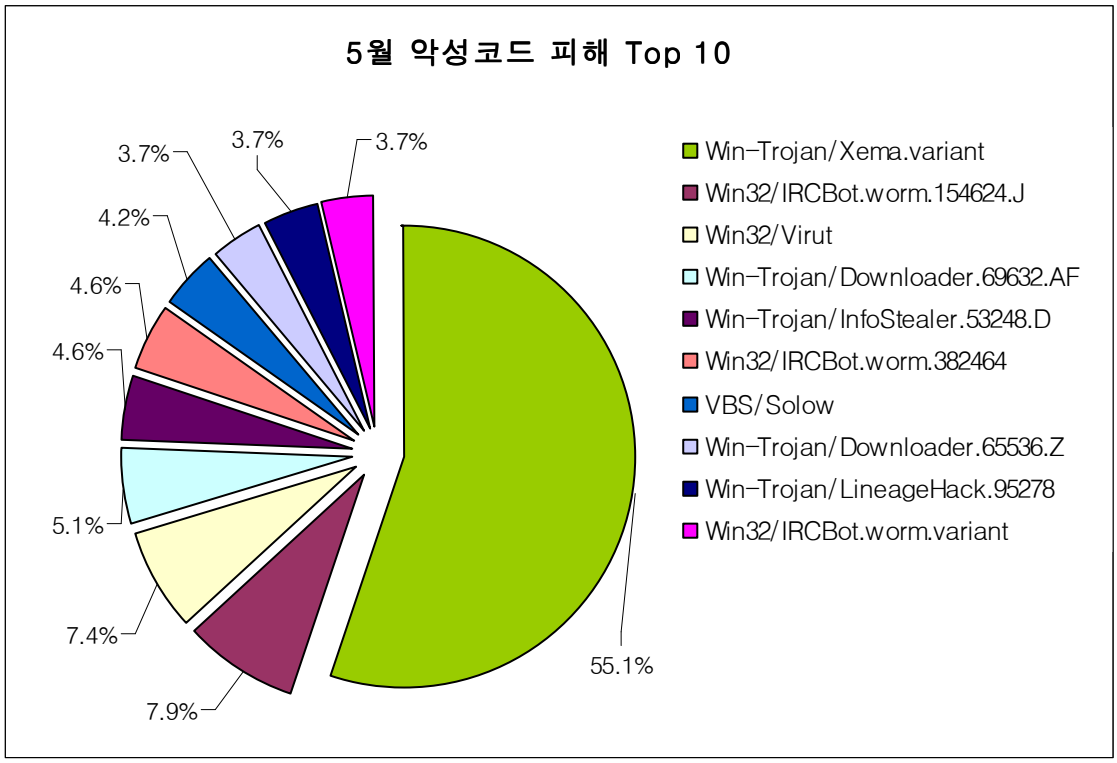
순위		악성코드명	건수	%
1	↑2	Win-Trojan/Xema.variant	119	55.1%
2	new	Win32/IRCBot.worm.154624.J	17	7.9%
3	↑5	Win32/Virut	16	7.4%
4	new	Win-Trojan/Downloader.69632.AF	11	5.1%
5	new	Win-Trojan/InfoStealer.53248.D	10	4.6%
5	new	Win32/IRCBot.worm.382464	10	4.6%
7	new	VBS/Solow	9	4.2%
8	new	Win-Trojan/Downloader.65536.Z	8	3.7%
8	new	Win-Trojan/LineageHack.95278	8	3.7%
8	↓4	Win32/IRCBot.worm.variant	8	3.7%
합계			216	100.0%

[표 2-1] 2007년 5월 악성코드 피해 Top 10

월 악성코드 피해 동향

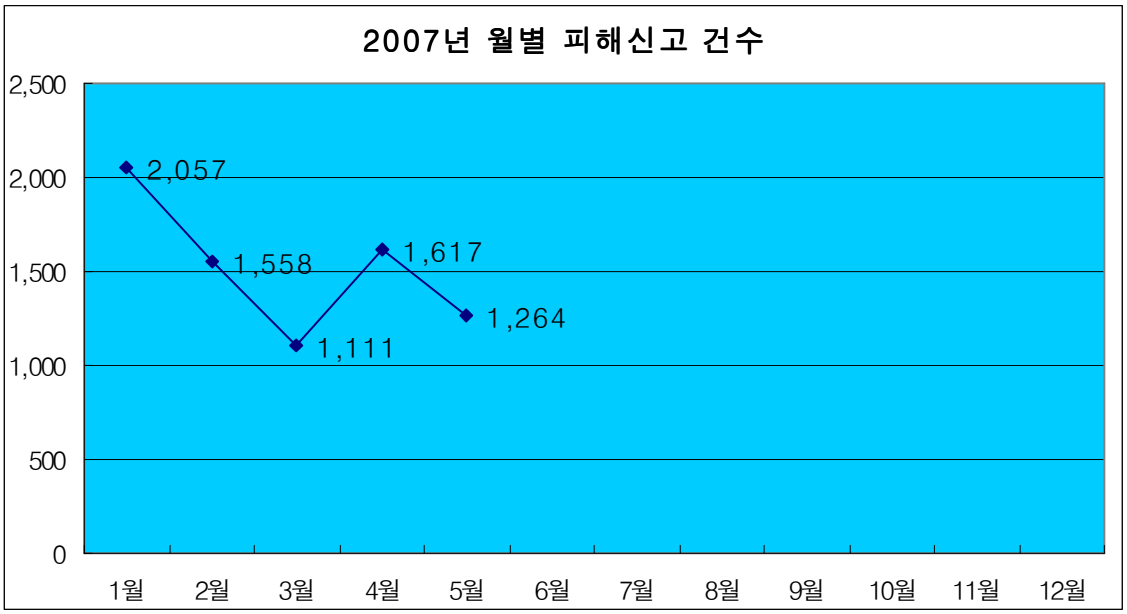
2007년 5월 악성코드 Top10에는 전월에 4위였던 아이알씨봇(Win32/IRCBot.worm.variant)이 8위로 하락하였으며, 전월의 1위였던 Win-Trojan/Downloader.38400.I은 Top10순위에서 밀려 났다. 바이럿(Win32/Virut)은 5위로 순위가 두 단계 상승하였다. 1월에서 4월까지 꾸준히 Top10중 7종을 차지하였던 트로이목마류가 5월에 이르러 Top10중 5종만 속하는 약세를 보였다. 5월 악성코드 1위는 Win-Trojan/Xema.variant이며, 지난달에 비해 한 단계 상승하였으며, 이외 순위에는 새로운 트로이목마 및 기타 악성코드가 Top10에 진입하였다.

5월의 악성코드 피해 Top 10을 도표로 나타내면 [그림 1-1]과 같으며, 1위인 Win-Trojan/Xema.variant 이외의 나머지 9개 악성코드들의 비율을 비교하면 근소한 차이임을 확인 할 수 있다.



[그림 1-1] 2007년 5월 악성코드 피해 Top 10

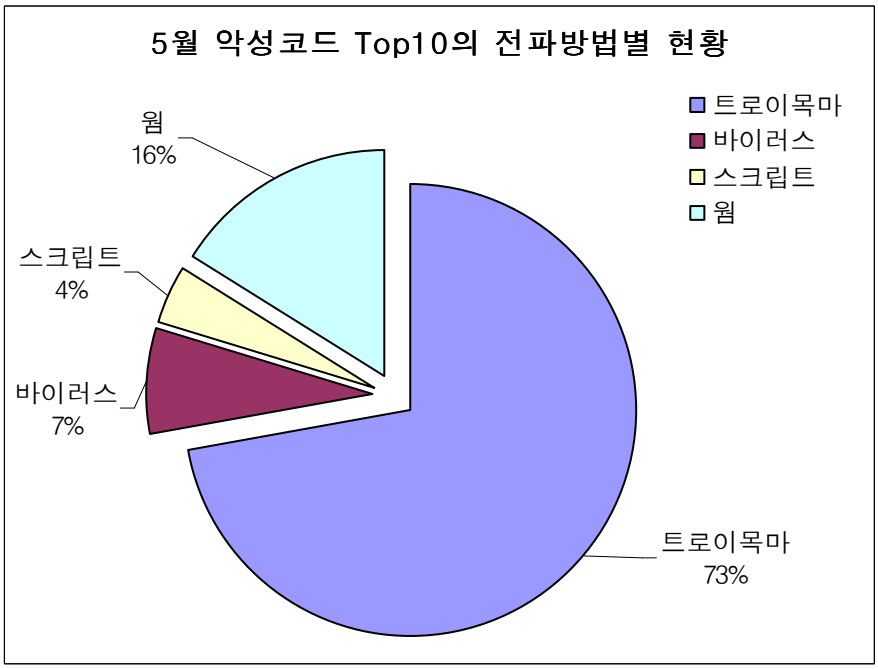
[그림 1-2]에서와 같이 1월부터 꾸준히 감소하던 피해신고는 4월에 증가하다가 5월이 되면서 다시 감소하였음을 알 수 있다. 잠시 증가하였던 4월에는 아이알씨봇(IRCBOT) 변형의 다수 출현이 피해신고가 증가한 원인이었고, 금월은 광범위하게 확산되는 웜(메스메일러, 봇)보다는 인터넷 사이트, 게시판으로 전파되는 트로이목마 위주로 확산됨으로써 피해건수가 감수되는 것으로 보인다.



[그림 1-2] 2007년 월별 피해신고건수

5월 악성코드 Top 10 전파방법 별 현황

[표 1-1]의 악성코드 피해 Top 10에서 확인된 악성코드는 [그림 1-3]과 같이 전파 방법을 기준으로 구분될 수 있다.



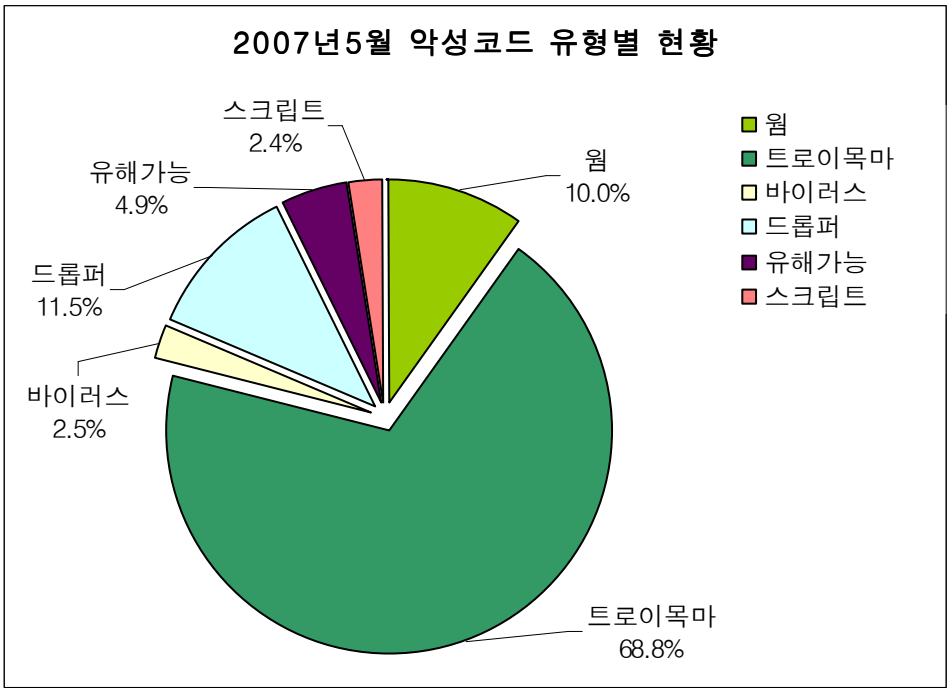
[그림 1-3] 2007년 5월 악성코드 Top 10의 전파방법별 현황

5월에도 변함없이 트로이 목마류가 가장 많은 피해를 발생시켰으며, 점유률은 73%로 전월(53%)에 비해 증가하였으며, 웜의 경우 전월(31%)에 비해 소폭 하락하였다. 바이러스는 바

이럿(Win32/Virut)의 순위가 두단계 올랐으나 점유율은 반대로 하락하였다.

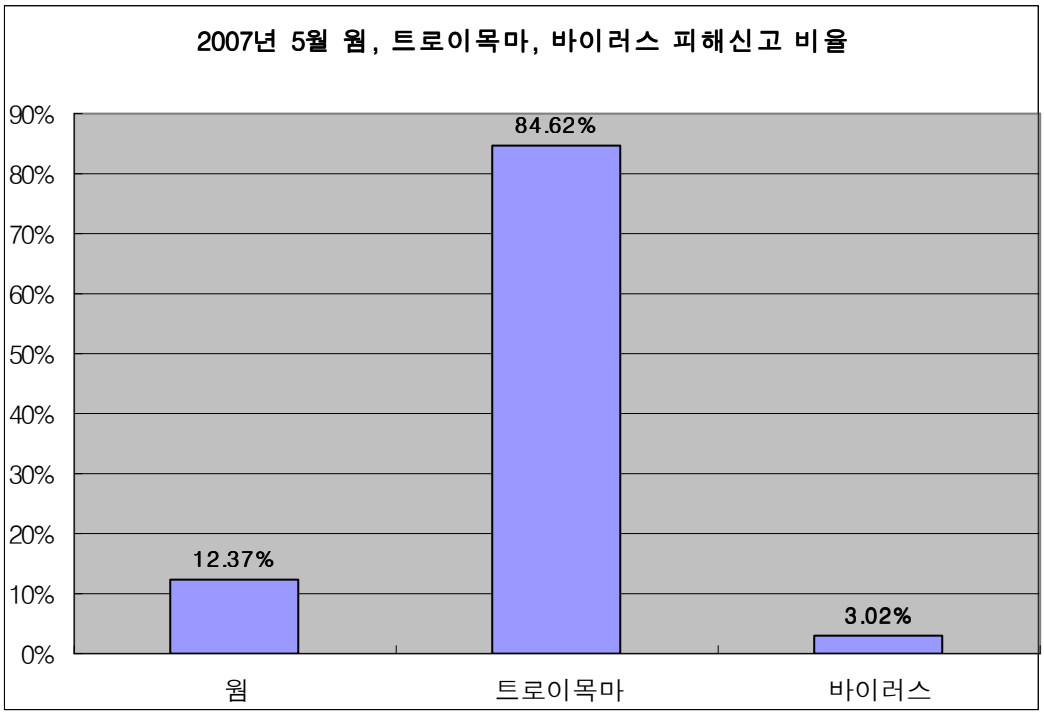
피해신고 된 악성코드 유형 현황

2006년 5월에 피해신고 된 악성코드의 유형별 현황은 [그림 1-4]와 같다.



[그림 1-4] 2007년 5월 피해 신고된 악성코드 유형별 현황

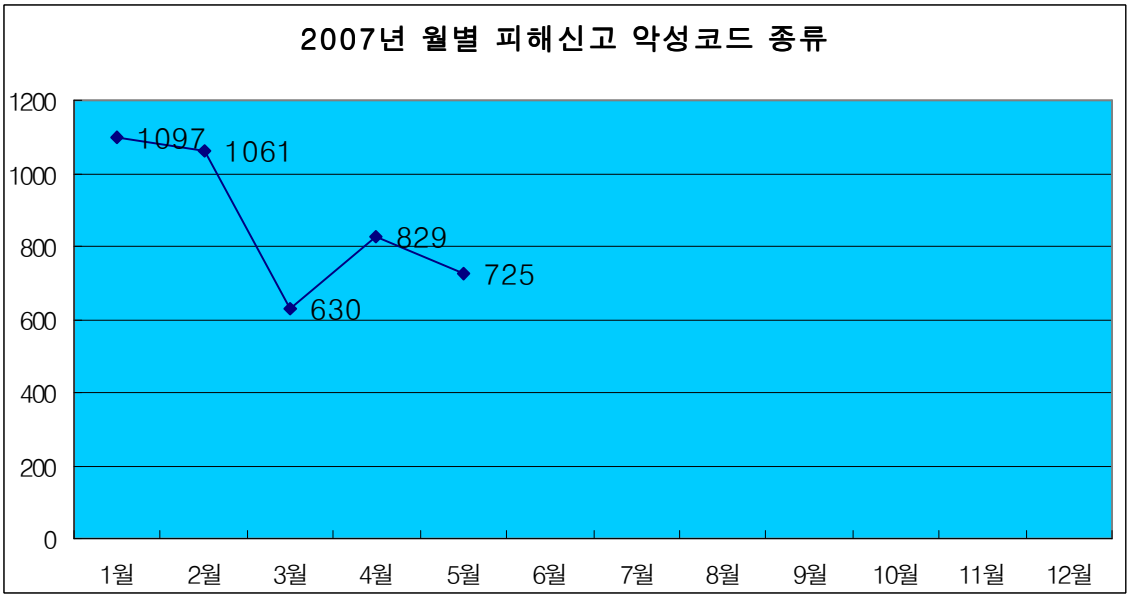
전체 피해 신고에서의 악성코드 유형을 확인해보면, 트로이목마가 68.8%로 가장 많았으며 그 다음은 드롭퍼가 11.5%였고, 3위는 웜으로 10%를 차지하였다. 그 외 유해가능프로그램이 4.9%, 뒤를 이어 스크립트가 2.4%, 바이러스는 2.5%였다. 이중 주요 악성코드 유형인 트로이목마, 바이러스, 웜에 대한 피해신고 비율을 따져보면 [그림 1-5]와 같다.



[그림 1-5] 2007년 5월 웜, 트로이목마 피해신고 비율

월별 피해신고 된 악성코드 종류 현황

[그림 1-6]에서와 같이 피해 신고된 악성코드 종류는 1월부터 3월까지 꾸준히 감소하다가 4월에 소폭상승하고 다시 5월에 이르러 다시 소폭 감소된 것을 알 수 있다.



[그림 1-6] 2007년 월별 피해신고 악성코드 종류 개수

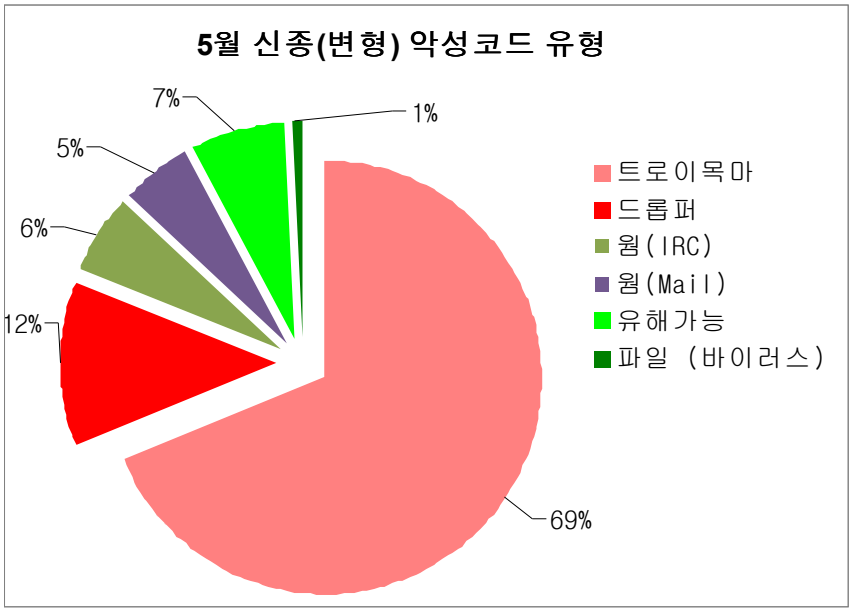
국내 신종(변형) 악성코드 발견 피해 통계

5월 한달 동안 접수된 신종 (변형) 악성코드의 건수는 [표1-2], [그림1-7]과 같다.

	웜	트로이	드롭퍼	스크립트	파일	매크로	부트	부트/파일	유해가능	비윈도우	합계
3 월	67	323	79	2	9	0	0	0	8	0	492
4 월	84	334	78	0	5	0	0	0	17	0	518
5 월	53	331	59	0	4	0	0	0	34	0	481

[표 1-2] 2007년 최근 3개월간 유형별 신종 (변형) 악성코드 발견현황

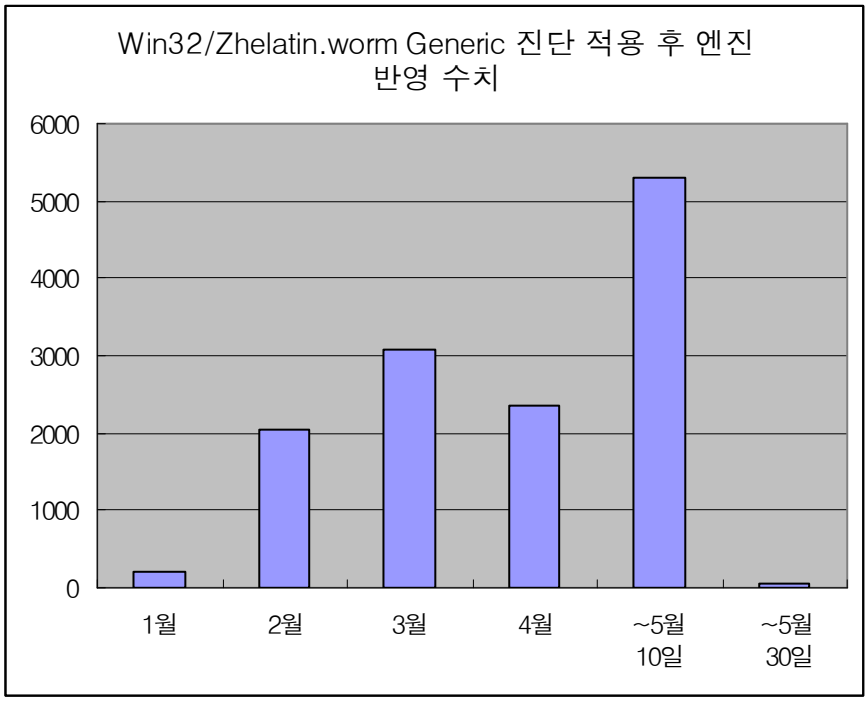
이번 달은 전월 대비 8% 정도 악성코드 수가 감소 하였다. 이는 지난달에 다수 발견되었던 Win32/IRCBot.worm (이하 악성 IRCBot 웜)과 Win32/Zhelatin.worm (이하 젤라틴 웜) 등의 감소로 기인한 것으로 분석된다. 악성 IRCBot 웜 경우 지난달 알려진 MS RPC DNS 서버관련 취약점을 이용한 변형이 증가하였다가 해당 취약점을 이용한 악성 IRCBot 웜이 크게 피해를 주지 못하자 변형 제작율이 떨어진 것으로 추정된다. 이는 취약점 사용으로 인한 확산 및 감염 성공률이 높다면 제작자들은 적극적으로 사용했을 것이라는 추정에 근거한다. 또한 젤라틴 웜의 감소는 이 웜의 제작자가 변형을 유포를 하지 않아 자연스럽게 감소하였다기보다는 V3 엔진에 젤라틴 웜이 사용한 다형성 루틴과 실행압축 형태를 진단 할 수 있는 Generic 한 진단 함수를 개발하여 엔진에 반영 한 결과로 보인다.



[그림 1-7] 2007년 5월 신종 및 변형 악성코드 유형

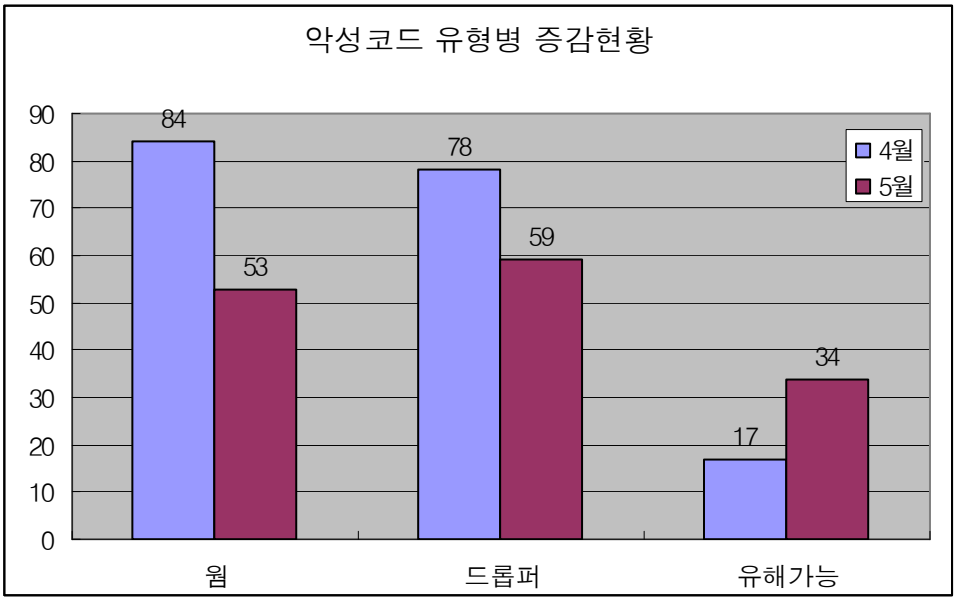
[그림 1-8]에서와 같이 젤라틴웜 변종에 대한 고객 접수 샘플 및 타사 샘플에 대한 엔진 반영 비율이 뚜렷하게 감소하였다. 물론 이를 기반으로 실제 샘플 수가 감소했다고 분석할 수는 없으나 본 리포트의 기초가 되는 통계자료는 안철수연구소가 고객으로부터 접수 받은 샘

플을 첫 번째 기준으로 작성하기 때문에 국내외 다른 업체 및 기관과의 악성코드 관련 통계와 차이가 발생 할 수 있다. 또한 악성코드에 대한 피해가 국지적으로 뚜렷하기 때문에 최근 들어 악성코드 통계를 타사와 비교하기에는 다소 무리가 있다. 해당 악성코드는 주로 국외(북미 또는 유럽지역)에서 피해가 많았으나, 국내에서는 국외에 비하여 뚜렷한 피해를 주지 않고 있다. 이는 해당 악성코드가 은폐형으로 일반 사용자들이 쉽게 감염 여부를 알아채지 못하기 때문에 피해나 샘플신고가 그다지 원할 하지 못하기 때문이다.



[그림 1-8] Win32/Zhelatin.worm Generic 진단 적용 후 엔진 반영 수치

[그림 1-9]와 같이악성코드 유형별로 웜과 드롭퍼 유형의 감소가 눈에 띄고, 유해가능 프로그램은 오히려 지난달과 비교하여 2배 증가한 수치를 보이고 있다.



[그림 1-9] 2007년 5월 감소 및 증가 악성코드 유형

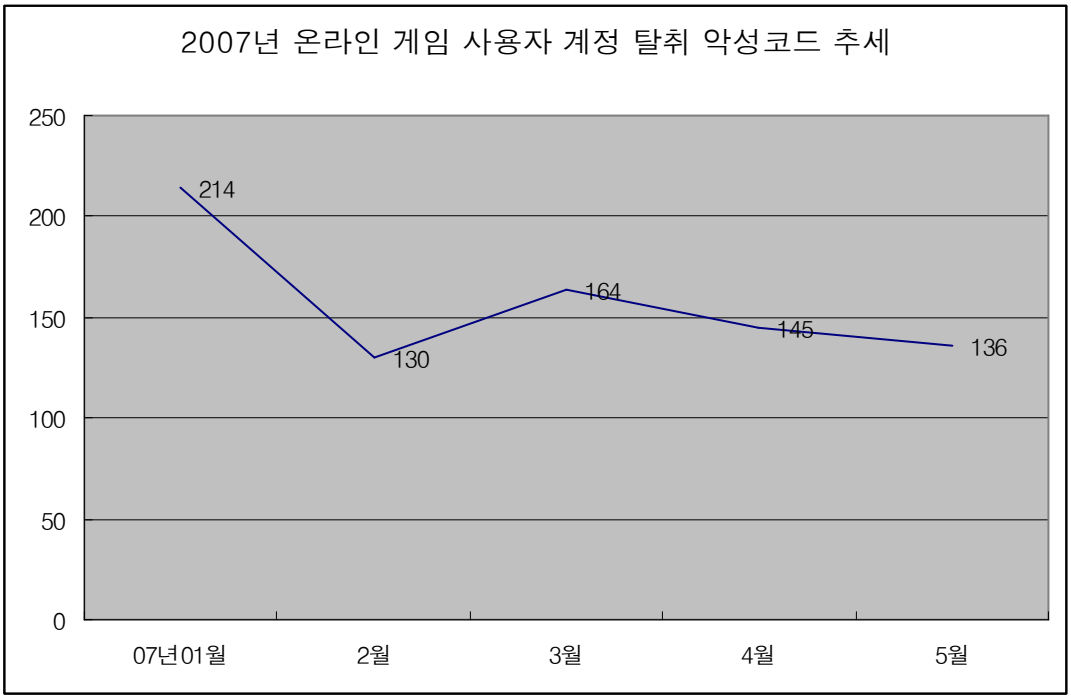
웜과 드롭퍼는 지난달 대비 37%, 24% 정도 하락 하였다. 드롭퍼의 감소 원인은 중국발 악성코드의 영향이 크다고 하겠다. 드롭퍼 대부분이 온라인 게임의 사용자 계정을 훔쳐내는 유형이 80% 이상 차지 하고 있으며 감소도 역시 해당 유형의 악성코드가 많았다. 따라서 드롭퍼에서 Drop 된 트로이목마의 수도 지난달과 비교하여 7% 정도 감소 하였다.

유해가능 프로그램의 경우 지난달에 언급 했던 Win-AppCare/Virtumond (이하 버추몬드)의 증가율이 뚜렷하다. 팝업광고를 노출 하는 증상이 있는 이 유해가능 프로그램은 악성코드로 분류되기도 하는데, 안철수연구소는 해당 악성코드의 진단율을 높이기 위해서 Generic 한 진단방법을 연구중에 있으며 조만간 그 결과를 토대로 엔진에 반영 할 예정이다.

실행 파일을 감염 시키는 바이러스는 이번 달에 4종이 발견 되었다. 모두 신종으로 다음과 같다.

- Win32/Alman: Win-Trojn원형과 B 형이 존재한다. B 형은 암호화된 바이러스 바디를 가지고 있으므로 진단을 위해서는 복호화 작업이 필요하다.
- Win32/Expiro: 감염된 파일은 *.IVR 이란 확장자로 변경해두며 이는 중복감염 여부를 체크 할 때 사용한다.
- Win32/Klest: 마지막 섹션에 652 바이트 만큼의 쉘코드를 추가한다. 해당 쉘코드는 특정 호스트로부터 파일을 다운로드 받도록 하는 코드가 담겨 있다.
- Win32/Mebangki: 후위형 바이러스로 감염된 파일은 .TNT 섹션이 추가 되며 14,516 바이트 증가한다.

다음 [그림 1-10]은 트로이목마 및 드롭퍼의 전체 비중에서 상당한 비율을 차지 하는 온라인 게임의 사용자 계정을 탈취하는 트로이목마의 추세이다.



[그림 1-10] 온라인 게임 사용자 계정 탈취 트로이목마 현황[1]

전월대비 7% 정도 감소율을 보이고 있다. 이러한 감소율은 2월을 제외하고 최근 3개월간 하락수치를 보이고 있다. 이는 다음과 같은 원인에 기인한 것으로 추정된다.

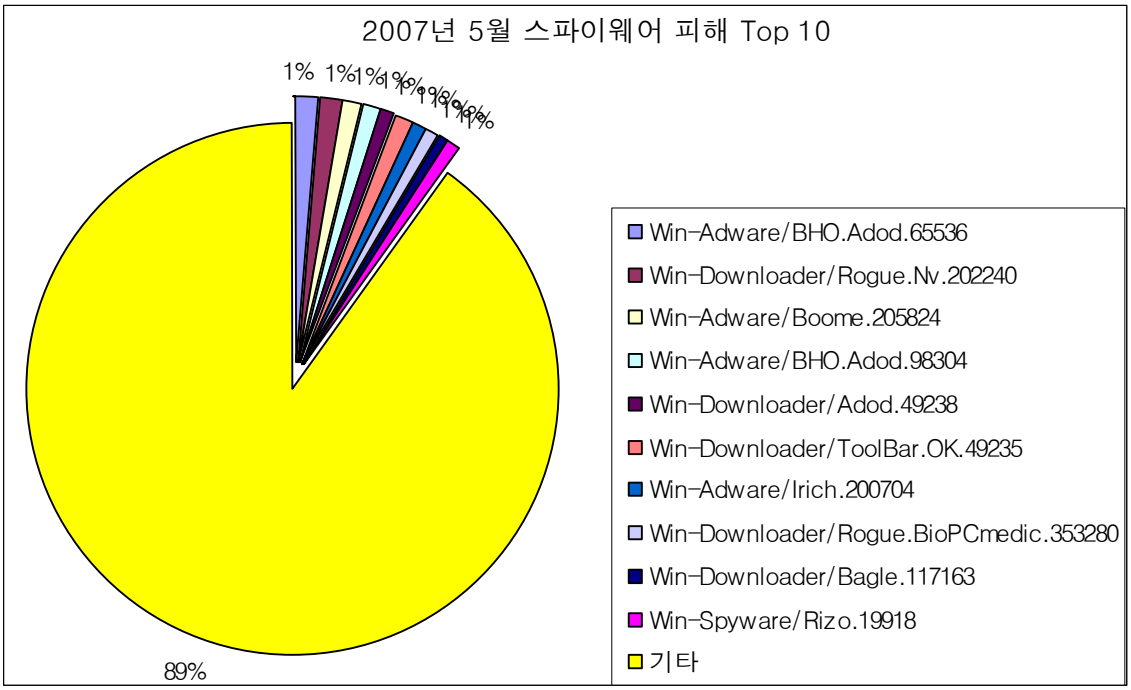
- 국내 온라인 게임 보안 솔루션의 고도화
- 대만 및 중국 현지 자국 온라인 게임 시장의 활성으로 타켓 게임 변경
- 중국내 주식시장 광풍으로 인한 악성코드 제작율 하락
- 계정 탈취로 인한 아이템 현금화 후 중국 및 대만발 전화 사기 업체 설립

위 정리 내용중 현재 중국 경제상황과 국내의 피해상황을 유추하여 보고 일부는 필자 나름대로 약간의 상상력을 추가해서 정리 한 것도 있다.

(2) 5월 스파이웨어 통계

순위		스파이웨어 명	건수	비율
1	New	Win-Adware/BHO.Adod.65536	9	1%
2	New	Win-Downloader/Rogue.Nv.202240	7	1%
3	New	Win-Adware/Boome.205824	7	1%
4	New	Win-Adware/BHO.Adod.98304	6	1%
5	New	Win-Downloader/Adod.49238	6	1%
6	New	Win-Downloader/ToolBar.OK.49235	6	1%
7	New	Win-Adware/Irich.200704	5	1%
8	New	Win-Downloader/Rogue.BioPCmedic.353280	5	1%
9	New	Win-Downloader/Bagle.117163	4	1%
10	New	Win-Spyware/Rizo.19918	4	1%
		기타	453	90.0%
합계			498	100%

[표1-3] 2007년 5월 스파이웨어 피해 Top 10



[그림 1-11] 2007년 5월 스파이웨어 피해 Top 10

2007년 최근 6개월 간의 스파이웨어 피해 통계를 살펴보면 피해 통계 상위 Top 10에 위치한 스파이웨어 중 그 어느 것도 다음 달 통계의 피해 통계 Top 10에 오르지 않았다는 사실을 알 수 있다. 이 점은 2007년 발견된 모든 스파이웨어가 꾸준한 피해를 입히고 있지 않다

는 사실을 말해준다. 2007년 발견된 스파이웨어 중 변형의 배포 주기가 짧은 스파이웨어 크립터(Win-Spyware/Cryprt)와 온라인게임 계정 유출 목적의 스파이웨어를 제외하고는 꾸준한 피해를 입힌 스파이웨어를 찾아보기는 힘들다.

2007년 5월 가장 많은 피해 신고 접수 건수를 기록한 애드웨어 어도드(Win-Adware/Adod.65536)는 허위 안티-스파이웨어 프로그램과 같은 다른 스파이웨어에 의해 사용자 동의 없이 설치되는 BHO(Browser Helper Object, 브라우저 도우미 개체) 형태의 애드웨어이다. 애드웨어 어도드가 설치된 시스템에서 IE 브라우저가 강제로 종료되는 증상은 IE 브라우저의 확장기능으로 동작하는 애드웨어 어도드의 자체 오류에 의한 것으로, 이 때문에 2007년 5월 7일 많은 피해 신고가 접수되었다. 애드웨어 어도드 이외에도 피해통계 Top 10의 대부분은 국내에서 제작된 애드웨어가 차지하고 있는 것도 5월 피해통계의 특징이다.

2007년 5월 유형별 스파이웨어 피해 현황은 [표 1-4]와 같다.

전체 스파이웨어 피해 신고 건수는 전월인 4월에 비하여 약 100건 정도 증가하였으며, 피해통계 Top 10의 내용을 반영하듯 전체 피해 통계 수치에서도 국내에서 제작/배포되는 애드웨어에 의한 피해 신고가 크게 증가하였다.

	스파이웨어류	애드웨어	드롭퍼	다운로더	다이얼러	클리커	익스플로잇	AppCare	Joke	합계
3월	123	100	25	69	1	14	6	0	0	338
4월	233	94	52	81	2	23	7	6	0	498
5월	320	109	22	122	2	10	2	9	0	596

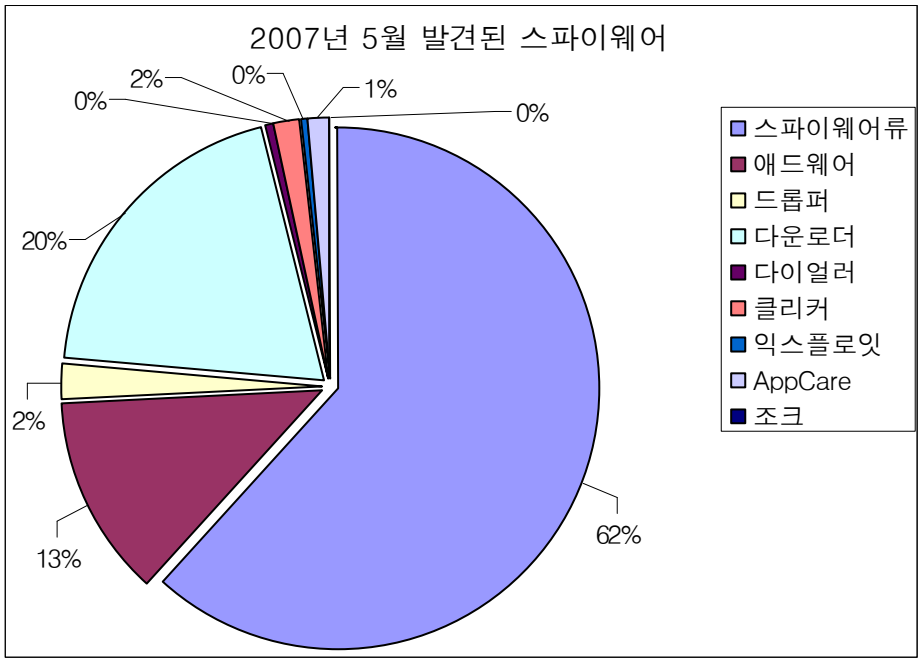
[표 1-4] 2007년 5월 유형별 스파이웨어 피해 건수

5월 스파이웨어 발견 현황

5월 한달 동안 접수된 신종(변형) 스파이웨어 발견 건수는 [표 1-5], [그림 1-12]와 같다.

	스파이웨어류	애드웨어	드롭퍼	다운로더	다이얼러	클리커	익스플로잇	AppCare	Joke	합계
3월	48	17	10	20	0	5	2	0	0	102
4월	105	20	13	30	1	5	3	3	0	180
5월	143	29	5	46	1	4	1	3	0	232

[표 1-5] 2007년 5월 유형별 신종(변형) 스파이웨어 발견 현황

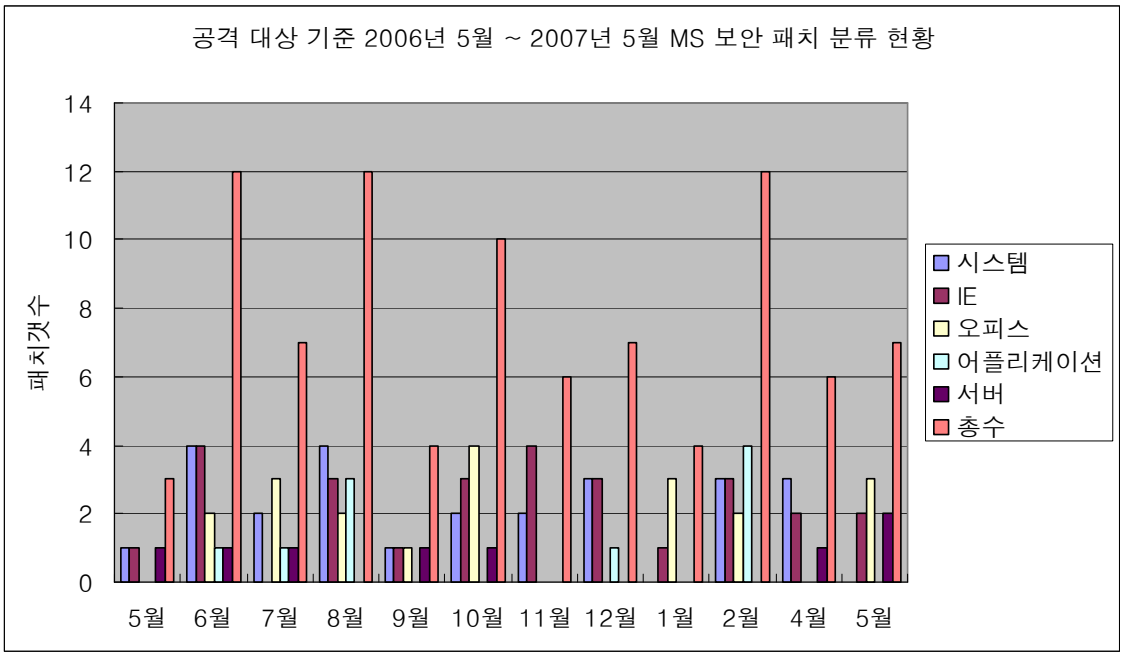


[그림 1-12] 2007년 5월 발견된 스파이웨어 프로그램 비율

[표 1-5]와 [그림 1-12] 2007년 5월 발견된 신종 및 변형 스파이웨어 통계를 보여준다. 4월에 비하여 약 29% 증가한 232건의 신종 및 변형 스파이웨어가 발견되었다. 최근 3개월간 꾸준히 새로운 변형이 발견되고 있는 스파이웨어 크립터(Win-Spyware/Crypter)의 경우 40건의 새로운 변형이 발견되었으며, 온라인 게임 계정 유출 목적의 스파이웨어도 약 30건의 변형이 발견되었다.

(3) 5월 시큐리티 통계

[그림 1-13]과 같이 2007년 5월에는 마이크로소프트사에서 총 9개의 보안 업데이트를 발표하고, 발표된 업데이트는 모두 긴급(Critical)에 해당된다. 이 중에서 오피스 취약점들(MS07-023, MS07-024, MS07-025)에 대한 패치가 포함되었으며, Exchange 서버 관련 취약점인 MS07-026, DNS 서버 관련 취약점인 MS07-029가 포함되어 있다.



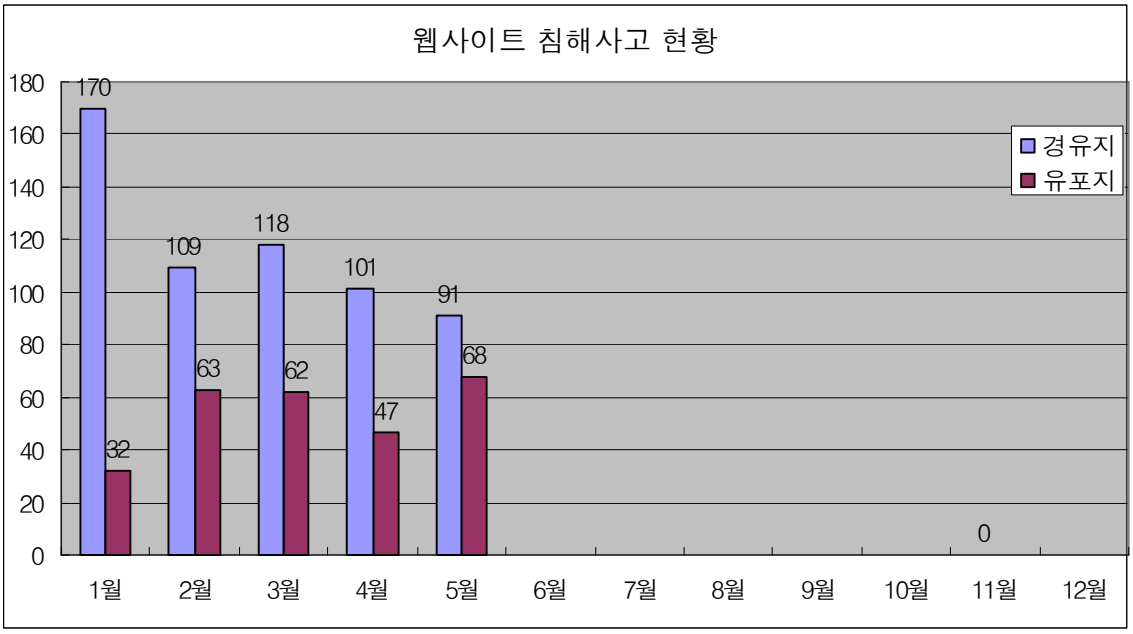
[그림 1-13] 2006년 5월 ~ 2007년 5월 공격대상 기준 MS 보안 패치 현황

[그림 1-13]을 보면, 전반적으로 2007년에 들어와서, 오피스 및 IE 취약점이 증가 추세에 있는 것을 알 수 있다.

2007년에 이슈가 되고 있는 마이크로소프트 취약점은 인터넷 익스플로러 VML 취약점인 MS07-004, Animated Cursor Handling 취약점인 MS07-017, 그리고 DNS 서버 취약점인 MS07-029 등이 있다. 그리고 몇몇 국내 ActiveX 관련 취약점도 발견되었다.

2007년에 들어와서도 오피스 취약점의 증가추세는 꾸준하다. 오피스 취약점 공격을 방지하기 위해서는 신뢰되지 않은 사이트 접속 및 오피스/아래 한글 파일이 메일로 첨부해서 오는 경우에 주의가 필요하며, 보안 패치를 반드시 해야한다. 아울러 Anti-Virus 제품 및 개인 방화벽 제품 또한 필요하다.

2007년 5월 웹 침해사고 현황



[그림 1-14] 웹사이트 침해사고 현황

2007년 5월의 침해/악성 코드 유포 사이트의 수는 91/68이다. 2007년 4월과 비교하여 침해 사이트의 수는 10개 감소하였지만 유포사이트의 수가 19개 증가하였다. 이는 악성코드 유포지를 다양하게 하여 보안장비나 Anti Virus 프로그램의 대응을 무력화하려는 시도로 분석된다. 악성코드 배포 유형은 2007년 4월과 마찬가지로 MS07-017(Ani) 취약점이 전체의 68%로 과반수 이상을 차지하고 있다. 이러한 현상은 MS07-017취약점 이후 Internet Explorer(IE)과 관련하여 새롭게 발견된 취약점이 없기 때문인 것으로 보인다. 앞으로 새로운 IE의 취약점이 발견되기 전까지는 MS07-017 취약점을 이용한 악성코드 유포하는 비율이 과반수 이상을 차지하는 현상이 계속하여 유지될 것으로 보인다.

II. ASEC Monthly Trend & Issue

(1) 악성코드 – Autorun.inf 파일의 정체에 대하여

이번 달 악성코드 이슈로는 먼저 윈도우 정품 인증을 가장한 트로이목마가 국외에서 보고 되었는데 일종의 사회공학기법을 이용한 형태였다. 그리고 새로운 형태의 SSDT 후킹을 시도하여 자신의 은폐 모듈을 숨기는 악성코드가 국, 내외에서 보고 되었다. 이 새로운 시도로 기존의 은폐형 악성코드 탐지툴로부터 자신을 우회하기 때문에 이목이 집중 되었다. 또한 최근 들어 가장 큰 이슈가 집중 되는 플래쉬 메모리에 생성 되는 autorun.inf 파일의 정체에 대해서 알아본다.

► 윈도우 정품 인증을 가장한 트로이목마

Win-Trojan/Kardphisher는 윈도우 정품 인증을 가장하여 사용자로 하여금 신용카드 정보를 입력 하도록 유도하는 트로이목마이다. 이 트로이목마는 마치 윈도우 정식 인증을 위하여 그럴듯한 메시지와 절차로 사용자로 하여금 신용카드 정보를 입력 받고 이것을 탈취한다. 무엇보다도 이 트로이목마는 실행 된 후 자신을 가장 상위로 활성화 하기 때문에 다른 응용 프로그램이나 화면으로 전환 할 수가 없게 된다. 개인정보와 카드입력을 받는 방법은 특별한 기법이 아닌 단지 Fake 된 화면을 보여주는 일종 사회공학기법에 불과 한다. 최근 MS 는 윈도우와 IE 7 에 대하여 정품 인증을 강화 하면서 이것을 노리고 사용자의 민감한 개인정보를 갈취하는 트로이목마가 출현 한 것으로 보인다.

[그림 2-1] Win-Trojan/Kardphisher 실행화면 1

[그림 2-2] Win-Trojan/Kardphisher 실행화면 2

► **새로운 형태의 SSDT 후킹을 시도하는 트로이목마**

Win-Trojan/Almanahe 이라고 명명된 이 악성코드는 기존과 다른 SSDT (System Service Descriptor Table) 후킹을 이용하여 기존의 알려진 루트킷 진단 프로그램에서 진단되지 못하도록 자신을 숨긴다. 이 트로이목마가 사용한 SDT 후킹 기법은 후킹 주체가 악성코드 모듈이 아닌 정상 ntoskrnl.exe를 가르킨다. 즉, 서비스 함수 포인터의 위치에 대해서 해당 트로이목마는 서비스 주체인 ntoskrnl.exe를 가르키도록 했다.

후킹 주체인 은폐된 커널 드라이버가 악성코드 자신이 아닌 ntoskrnl.exe를 보이도록 해서 우회한다. 그러나 후킹된 함수까지 숨기지는 못하므로 이를 통하여 SSDT 후킹 여부는 확인될 수 있다.

Id	Service Name	Hooked	Address	Module
25	NtClose	Yes	0x8054EDC1	C:₩WINDOWS₩system32₩ntoskrnl.exe
63	NtDeleteKey	Yes	0x8054EDD3	C:₩WINDOWS₩system32₩ntoskrnl.exe
65	NtDeleteValueKey	Yes	0x8054EDD9	C:₩WINDOWS₩system32₩ntoskrnl.exe
71	NtEnumerateKey	Yes	0x8054EDBB	C:₩WINDOWS₩system32₩ntoskrnl.exe
97	NtLoadDriver	Yes	0x8054EDB5	C:₩WINDOWS₩system32₩ntoskrnl.exe
145	NtQueryDirectoryFile	Yes	0x8054EDC7	C:₩WINDOWS₩system32₩ntoskrnl.exe
207	NtSaveKey	Yes	0x8054EDCD	C:₩WINDOWS₩system32₩ntoskrnl.exe

[그림 2-3] Win-Trojan/Almanahe 의 SSDT 후킹 함수 및 주체

이 트로이목마는 실행 파일을 감염시키는 바이러스 증상도 가지고 있으며, 또한 일부 온라인 게임의 사용자 계정을 훔쳐내는 증상도 있다. V3 는 이 또한 Win32/Alman 이라고 명명했고 진단 / 치료가 가능하다.

► 플래쉬 메모리에 생성된 Autorun.inf 의 정체는?

근래 들어 이동식 드라이브 (대부분 USB 방식의 플래쉬 메모리 스틱)에 생성된 Autorun.inf 의 정체에 대한 문의가 다수 접수되고 있다.. 매체에 대한 단가 하락과 대량 생산은 자연스럽게 이러한 미디어에 대한 접근을 쉽게하므로 요즘 플래쉬 메모리 스틱을 한 개 이상은 보유하고 있다고 해도 과언이 아니다. 따라서 이러한 미디어를 대상으로 악성코드를 감염시키려는 악성코드 제작자들의 노력이 계속되고 있다.

그렇다면 왜? 이동식 드라이브를 감염 대상으로 하는 것일까? 이는 예전에 플로피 디스켓에 부트 바이러스나 파일 바이러스를 감염시켰던 것과 같이 좀 더 확산력을 높이기 위해서이다. 또한 그 당시에는 없었던 Autorun.inf 파일을 이용하여 자동으로 실행할 대상을 지정하여 악성코드를 사용자 의도와는 관계없이 실행 및 감염 그리고 확산 시키려는데 목적이다.

VBS/Solow는 Autorun.inf 파일을 생성하는 대표적인 악성코드이다. 이외에도 스크립트 형태가 아닌 *.EXE 확장자를 갖는 실행 파일 형태의 악성코드도 있다. VBS/Solow는 각 드라이브 루트 폴더 비롯하여 이동식 드라이브 루트 폴더에 Autorun.inf 파일을 생성한다. 이러한 활동을 200초 마다 반복적으로 실행 하기 때문에 사용자가 Autorun.inf 파일을 삭제해도 다시 생성된다.

생성된 Autorun.inf 파일은 특정 드라이브 또는 이동식 드라이브에 생성된 악성코드 복사본을 실행하도록 하는 명령이 포함되어 있다. 예를 들어 플래쉬 메모리 스틱에 Autorun.inf 파일이 있다면 해당 플래쉬 메모리를 USB 포트에 연결한 후 바탕화면의 내 컴퓨터를 선택하여 플래쉬 메모리 스틱이 연결된 이동식 드라이브를 클릭 할 경우 Autorun.inf 에 의해서 악성코드가 자동으로 실행된다.

고객들이 많이 질문하는 내용은 크게 2 가지로 다음과 같다.

- Autorun.inf 파일의 삭제후 재생성
- Autorun.inf 파일과 악성코드 파일을 수동으로 삭제한 경우 바탕화면에서 내 컴퓨터를 이용하여 각 드라이브 접근불가

먼저 Autorun.inf 파일의 재생성은 위에서 언급 한 것처럼 VBS/Solow 는 200초 마다 반복적으로 각 드라이브에 대한 Autorun.inf 파일을 생성한다. 또한 해당 악성코드는 스크립트 웜으로 자신이 실행되기 위해서 일종의 인터프리터인 WScript.exe 라는 파일을 실행 함으로써 자신의 스크립트를 실행한다. 그러므로 프로세스에서 실행중인 WScript.exe를 종료하지 않으면 Autorun.inf파일이 재생성되는 원인이 된다. 중요한 것은 WScript.exe는 정상적인 윈도우 파일이므로 삭제해서는 안된다. 두 번째로 Autorun.inf 파일과 악성코드를 사용자가 직접 삭제한 경우 내 컴퓨터를 이용한 각 드라이브 접근시 다음 [그림 2-4]와 같은 에러 메시지가 나오고 드라이브 열기가 불가능한 경우가 발생할 수 있다.

[그림 2-4] VBS/Solow 레지스트리 미치료시 나오는 에러 메시지

이는 자동 실행 되기 위해서 각 드라이브에 대한 클래스 ID가 저장된 레지스트리 키를 변경해주지 않았기 때문이다. 이 키는 각 드라이브의 클래스 ID 가 저장된 하위 키에 Autorun.inf 에 의해서 자동 실행되도록 대상 파일이 기록되어 있다. 따라서 이 하위 키를 변경 또는 삭제하지 않으면 드라이브 열기를 시도할 때마다 레지스트리 키 값에 명시된 파일이 존재하지 않는다는 메시지를 출력하고 열기가 불가능 해진다. 물론 탐색기를 통해서나 다른 파일 관리자를 이용해서는 각 드라이브 탐색은 가능하다.

► 구글 Adsense 의 부정 클릭 유도한 사건

5월 중순 MSN 메신저로 다음과 같은 메시지가 국내외 퍼졌다.

www.whoadmit(제거됨).com <- Find out who deleted and blocked you from the MSN

위 링크를 클릭하여 자신의 MSN 메신저 아이디와 비밀번호를 입력하면 자신을 버디 리스트에서 차단하거나 삭제한 것을 알려준다고 한다. 하지는 이는 사용자를 속이는 것이고, 입력 받은 계정과 비밀번호를 이용하여 MSN 서버로부터 버디 리스트를 받아와 위 링크를 현재 온라인된 모든 사용자에게 발송하는 사건이 있었다. 초기에는 이 링크를 보내는 별도의 악성코드가 있다고 추정하였지만, 실제로는 그렇지 않고, 위 링크의 주소로 들어가 MSN 계정을 입력하면 현재 로그인 된 자신의 MSN 메신저는 로그아웃이 되고, 입력 받은 계정으로 MSN 서버로 로그인하여 버디 리스트를 가져와 위 링크를 보내는 것으로 최종분석 되었다.

또한 해당 링크의 페이지는 구글의 Adsense 광고가 다수 노출되어 있었다. 이러한 점을 종합해 보면 위 링크의 방문자들로 하여금 구글 Adsense 의 부정 클릭을 유도하여 돈을 벌어드리려는 것으로 위와 같은 웹 페이지와 호스트를 운영하는 것으로 추정된다. 만약 위 링크에서 자신의 MSN 계정을 입력 했다면 혹시 있을지 모르는 개인 정보 도용을 예방하기 위해서라도 MSN 계정의 비밀번호를 지금 변경할 것을 권장한다.

(2) 스파이웨어 - 국산 스파이웨어의 증가

Win-Adware/Adod

5월 초, 다수의 고객으로부터 Internet Explorer(이하 IE)가 동작하지 않는다는 신고가 접수되어 인터넷 대란이 또다시 일어나는 것은 아닌지, 많은 사람들을 우려케 하였지만 다행히 특정 애드웨어가 설치된 PC에서만 발생하는 문제로 밝혀졌으며 그 주범은 Win-Adware/Adod였다.

Win-Adware/Adod가 PC에 설치되면 BHO(Browser Helper Object)와 Toolbar가 등록된다. 본래의 동작은 IE의 주소표시줄에 입력되는 한글 키워드를 감시하여 특정 한글 키워드가 입력되면 제작사와 제휴된 검색사이트의 검색결과를 노출하도록 되어있다. 그러나 BHO로 등록되는 모듈이 버그를 포함한 채로 배포되었고, IE가 시작되면서 BHO 모듈을 로딩할 때, Win-Adware/Adod의 잘못 제작된 BHO 모듈이 로딩되면서 오류가 발생하여, 이로 인하여 IE가 실행되자마자 종료되었다.

대부분의 애드웨어는 Win-Adware/Adod와 같은 문제점을 시한 폭탄처럼 안고 있다고 볼 수 있다. 애드웨어 제작업체들 대다수가 프로그램 개발시 간단히 동작만 확인할 뿐, 별도의 품질 테스트 과정을 거치지 않기 때문에 언제 어느 시점에서 시스템에 치명적인 오류를 야기할지 모르며 이와 같은 사례는 허다하다.

이러한 문제는 시스템의 중요 부분과 관련이 있을 경우 더욱 큰 문제가 될 수 있는데, Win-Adware/Adod와 같이 IE의 중요한 프로그램의 일부 모듈로 동작하거나 시스템 드라이버로 등록되는 경우 시스템을 망가뜨리거나 주요 어플리케이션 사용에 불편을 주는 등 사용자에게 큰 불편을 줄 수 있다.

특히 시스템 드라이버로 등록되는 경우는 BSOD(Blue Screen of Death)를 발생시키거나 윈도우가 부팅되지 않는 상황까지 발생할 수 있다. 과거 Win-Adware/Rogue.CC가 등록한 루트킷 드라이버로 인하여 레지스트리에 관련한 작업시 매번 BSOD가 발생한 사례가 있는데, 이 경우 사용자는 일반적인 방법으로는 Win-Adware/Rogue.CC를 제거할 수 없어 더욱 큰 문제가 되었다.

스파이웨어나 애드웨어 제작자들에게 시스템의 오류를 발생하지 않도록 테스트를 잘 해달라고 요구할 수는 없는 노릇이기에, 현재로서는 이러한 것들이 설치되지 않도록 사용자가 주의하거나 안티-스파이웨어 프로그램을 사용하여 이들을 제거하는 수 밖에 없다.

국산 스파이웨어 피해 급증

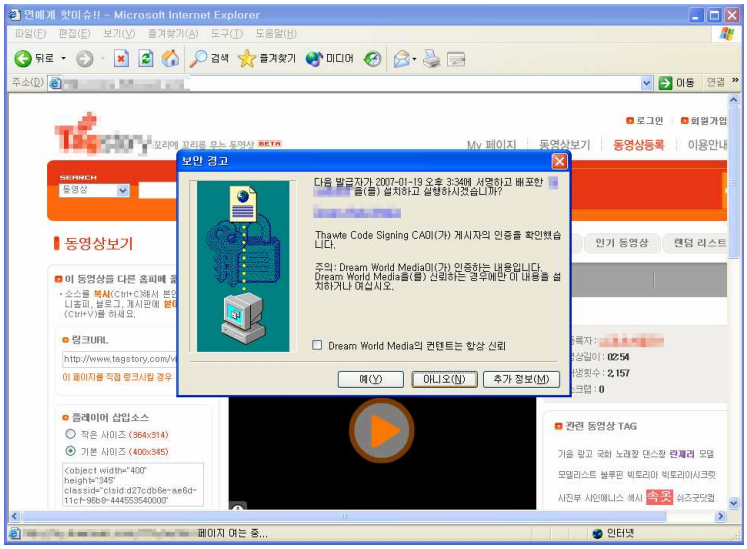
최근 국산 스파이웨어로 인한 사용자 피해 신고가 크게 증가하였다. 발견되는 스파이웨어의 수는 국외에서 제작된 것이 훨씬 많으나 그 피해 신고는 국내에서 제작된 것들로 인한 피해 신고가 더 많아지고 있다. 외국산 스파이웨어의 경우, 동일한 스파이웨어 대해 피해 신고가 다수 접수되는 것은 보안 취약점을 이용하거나 바이러스 혹은 웜에 의해 배포되어 그 확산력이 컸기 때문인 반면, 국산의 경우는 난이도가 높은 해킹 기법은 사용되지 않았으나 그 피해 신고는 비슷하거나 오히려 많았다. 물론 지역적인 이유로 그런 것일 수 있겠지만 유독 국산 스파이웨어로 인한 피해가 증가한 이유는 무엇일까?

이는 국내 UCC 산업의 발전과 관련이 있는 것으로 보인다. 과거, 카페나 게시판 등에 단순히 ActiveX 코드를 삽입하는 것이 전부였다면, 요즘은 블로그를 이용하여 UCC 동영상을 보기 위해서는 ActiveX 컨트롤을 설치해야 한다는 식으로 사용자들을 속이고 있다. 일부 UCC 동영상 사이트에서 동영상 재생을 위해서는 ActiveX를 설치해야 한다는 점을 교묘히 이용한 것이다.

국내 유명 포털 사이트를 보면 대부분 메인 페이지에 “실시간 인기 검색어”라는 제목으로 사용자들이 많이 입력하는 키워드의 목록이 나오는데, 이 키워드로 검색을 하면 관련된 UCC 동영상을 볼 수 있다는 제목의 글을 어렵지 않게 찾을 수 있다. 그런데 이 글들 중에 일명 “낚시글”이라고 불리는 사용자를 속이는 글들이 함께 노출된다.

스파이웨어 배포자들은 더욱 많은 사람들에게 자신의 사이트가 노출되도록 하기 위해 “실시간 인기 검색어”로 등록된 키워드가 들어간 글을 무작위로 생성하고 검색 사이트에 노출되기만을 기다린다. 그리고 국내 유명 검색 사이트에서는 보다 많은 UCC 확보를 위해 기계적으로 이를 검색 DB에 저장하고 결국 사용자에게 보여지는 것이다.

이러한 낚시글을 클릭해보면 UCC 동영상 관련 사이트인 것처럼 사용자들을 완전히 속이고, ActiveX 컨트롤을 설치할 것을 요구한다. 아래의 [그림 2-5]는 UCC와는 전혀 관련이 없으며 단순히 이미지로만 제작된 허위 UCC 동영상 사이트이다. 그림의 동영상 재생기 역시 가짜이다. 설치되는 ActiveX 컨트롤은 동영상 재생과는 전혀 관련이 없으며 설치할 경우 다수의 스파이웨어가 설치된다.



[그림 2-5] 허위 UCC 동영상 사이트

보다 많은 UCC 사이트를 검색하려는 검색 사이트의 헛점을 스파이웨어 배포자들이 잘 이용하고 있는 샘이다. 물론 검색 사이트에서 이러한 사이트를 발견하는 즉시 차단하는 등의 조치를 취하고 있는 것으로 보이지만 아직 미미하며, 이로 인한 피해는 지속적으로 증가하고 있는 실정이다.

(3) 시큐리티 - 디지털 환경에 따른 트렌드의 변화

마이크로소프트 사에서 이번 2007년 5월에 발표한 보안 업데이트는 총 7개로 모두 긴급(Critical)에 해당하는 업데이트들이다. 이중 DNS 서버의 RPC 원격코드 실행 취약점은 지난 4월 패치가 공개되기 이전에 공격코드가 공개되어 있었던 만큼, DNS 시스템을 운영하는 사용자는 바로 패치를 적용할 것을 권고한다. 이것은 DNS 서버 서비스에 바인딩 되어 있는 RPC 에 조작된 공격 패킷을 보내 임의의 코드를 실행할 수 있고 이미 IRCBot 악성코드에서 해당 코드를 사용하여 공격하는 것이 확인된 만큼 주의가 필요하다.

다음은 악의적인 공격에 이용될 수 있는 원격 코드 실행 취약점과 살펴볼 만한 주요 취약점들에 대한 목록이다.

위험등급	취약점	PoC
긴급	마이크로소프트 엑셀 원격 코드 실행 취약점 (MS07-023)	무
긴급	마이크로소프트 워드 원격 코드 실행 취약점 (MS07-024)	무
긴급	마이크로소프트 오피스 원격 코드 실행 취약점 (MS07-025)	무
긴급	윈도우 DNS 서버의 RPC 원격 코드 실행 취약점 (MS07-029)	유
긴급	Samba 원격 코드 인젝션 취약점	무
긴급	Sun JDK(Java Development Kit)이미지 처리 취약점	유

[표 2-1] 2007년 5월 주요 MS 취약점 패치

유닉스 기반에서 많이 사용되고 있는 삼바(SAMBA)는 윈도우의 파일 공유와 같은 기능을 제공해 주고 있는 프로그램으로서 사용자의 입력 파라미터로 /bin/sh 와 같은 인자가 넘어올 경우 원격지에서 명령어 실행이 가능하다. 삼바 3.0.0 – 3.0.25rc3 버전을 사용하고 있는 사용자는 최신의 버전으로 사용할 것을 권고하며 관련정보는 삼바 사이트 (http://www.samba.org) 에서 얻을 수 있다.

데이터의 분실 사건 사고의 현장

5월은 다른 달과 달리 데이터 분실 사고 등이 많이 발생하였다. 데이터 분실은 기업 또는 개인의 입장에서 큰 피해를 줄 수 있는 부분중의 하나이다. 특히 많은 자료들이 이제 디지털화 되어가며 과거에 몇 백 페이지 종이들이 유출되어야 했다면 이제는 작은 이동형 장치에 더 많은 정보들이 유출될 수 있는 환경에 놓여있다. 최근 해외 투자사로 유명한 JP 모건이 고객과 직원의 개인 정보가 들어 있는 백업 테이프를 분실한 사건이 발생하였다. 백업 테이프를 이동하는 과정에서 분실된 것으로 알려지고 있는데 대략 47,000 명 정도의 고객이 포함되었다고 한다. 이외 미국교통보안국(TSA:Transportation Security Administration)이 대략 100,000 여명의 직원 개인정보가 담긴 하드 드라이브를 분실한 것으로 알려졌다. TSA 직원

의 2002년 1월부터 2005년 8월 사이의 직원정보들로 이름, 사회보장번호, 은행 계좌 정보 등이 포함되었다고 하며 이메일을 통해 직원들에게 이 소식을 알렸다고 한다. 이 드라이브가 TSA의 본사에서 분실된 것으로 보고 있으며 FBI 와 미국 정보국에서 이 사건을 조사중이라고 한다. 이렇게 한 순간에 많은 양의 데이터가 유출될 수 있는 만큼 기업들은 이러한 데이터 분실에 대해서도 대비책을 세워두어야 한다.

웹 공간의 10% 가 위험하다고요?

여러분들이 방문하고 있는 웹 사이트는 항상 안전할까요? 방문한 사이트가 신뢰할 수 있으며 악의적인 요소가 포함되어 있지 않은 페이지로 만들어져 있다고 믿고 있었다면 이제 다시 생각해 보아야 할 것이다. 세계적 인터넷 검색업체 중 하나인 구글에서 흥미로운 발표를 하였는데, 4백5십만개의 웹 페이지를 분석한 결과 그 중 10%가 악의적 코드를 포함하고 있었다는 것을 발표하였습니다.

안철수연구소 시큐리티대응센터에서도 국내의 웹 해킹을 추적, 검토 등을 통해서 상당히 많은 수가 감염되어 있을 것으로 추정하고 있었지만 구글의 자료가 이를 뒷받침 하는 증거가 되었다. 10%인 45만 여개의 웹 사이트가 악의적 코드를 설치하는데, 이는 주로 사회 공학적 기법으로 사용자를 유혹할 만한 제목의 링크로 유도를 한다. 예를 들면, 포르노사이트 또는 유명 소프트웨어 다운로드 등이 해당된다.

이러한 악의적 코드의 설치는 주로 마이크로소프트사의 인터넷 익스플로러의 취약점을 이용하는 것으로 밝히고 있다. 그렇다면 왜 IE 의 취약점을 주로 이용하는 것인가? 이유는 바로 IE 의 사용률이 전세계적으로 가장 높기 때문이다. 많이 사용되지 않는 브라우저의 취약점보다는 많이 사용되는 브라우저 취약점을 이용하는 것이 보다 감염을 쉽게 시키기 위한 방법이기 때문이다. 악의적 코드 설치로 인한 피해는 즐겨찾기의 변경, 툴바 설치, 브라우저의 시작 페이지 변경 등이 해당되며 키로거 등의 설치를 통해 사용자 계정 정보를 빼내가기도 한다. 웹 환경의 변화에 따라 전통적인 방법의 악성코드 감염에서 이제는 웹으로 그 방법이 많이 옮겨지고 있다. 이에 대한 대책으로는 사용하는 컴퓨터에 최신의 보안 패치 설치와 의심되는 사이트는 방문하지 않고 안티바이러스, 개인용방화벽등과 같은 보안제품의 설치를 통해 피해를 최소화 할 수 있다.

오피스 취약점을 파헤쳐 본다.

이번 달에도 오피스 관련 많은 취약점(MS07-023, MS07-024, MS07-025)이 발생하였는데 오피스 취약점은 무엇이고 이를 통해 어떠한 피해들이 발생할 수 있는지 살펴보도록 한다. 사용자들이 많이 묻는 주요한 질문 몇 가지에 대해서 알아볼 것이다.

1. MS 오피스 취약점이란 무엇인가?

오피스 프로그램은 대다수 사용자가 이용하는 응용 프로그램으로 스프레드 시트 프로그램인 엑셀(Excel), 문서 작성/편집 프로그램인 워드(Word), 프리젠테이션 관련 프로그램인 파워포인트(PowerPoint), 데이터 베이스 관련 프로그램인(Access), 이메일 프로그램인 아웃룩(OutLook)등으로 구성되어 있다. MS 오피스 취약점은 이러한 오피스 프로그램 및 오피스 라이브러리에 버그(Bug)가 존재하는 것을 말한다.

사용자가 악의적으로 조작된 오피스 파일(File)을 읽는 과정에서, 사용자가 관리자 권한으로 로그인 되어 있는 경우 취약점을 악용한 공격자는 프로그램의 설치, 보기, 변경, 데이터 삭제 등과 같은 권한을 얻게 되어 시스템을 완전히 제어할 수 있게 된다. 하지만, 취약점을 이용한 공격에 성공하기 위해서는 사용자의 개입이 필요하다. 그러나, MS 오피스 프로그램이 기업의 많은 컴퓨터에 설치되어 있어 위험의 심각도가 높다고 볼 수 있다.

2. MS 오피스 취약점 동향 및 피해사례

MS 오피스 취약점은 2006년 상반기부터 본격적으로 나타나기 시작하였다. MS 사의 보안 패치 중에 2006년과 2007년 5월까지 MS 오피스 공격에 이용될 수 있는 취약점은 총 22건이다. 이것은 같은 기간 동안의 전체 보안 패치중 약 21.5% 정도를 차지하고 있다.

취약점을 이용한 공격에는 조작된 파일을 특정/불특정 사용자에게 메일 또는 웹으로 전달하여 사용자가 해당 오피스 파일(File) 읽는 경우 임의의 코드 또는 악성코드를 실행할 수 있게 된다.

악성코드는 V3 진단명으로 PP97M/Exploit-PPDropper, X97M/Exploit.Excel, X97M/Exploit.ControlExcel 등이 존재하며, 내부 코드에 트루잔(Trojan) 및 다운로더(Downloader)등이 포함되어져 있기도 하며, 최근에는 특정 오피스 취약점을 공격하는 자동 제작기가 중국에서 발견되기도 하였다.

외국뿐만 아니라 국내에서도 MS 오피스 취약점을 이용한 공격이 발생하고 있는데, 이러한 공격은 주로 특정 목적을 가지고 수행되는 것으로 보이며, 개인 및 기업등의 민감한 정보를 노리는 것으로 파악된다. MS 오피스 취약점은 제로 데이(Zero-Day) 공격에도 자주 사용이 되고 있기 때문에, 주의가 필요하다.

3. MS 오피스 종류에 따라 취약점이 발견되는 것인지?

엑셀, 워드, 파워포인트 개별로 취약점이 발견되며 또한 오피스 공통 라이브러리에서 취약점

이 발견되는 경우도 존재하며 공격형태는 유사하다고 볼 수 있다.

4. 사용자가 주의해야 할 점

1) 오피스 프로그램의 보안 패치를 주기적으로 해야 한다.
2) 오피스 파일을 메일 또는 웹으로 받은 경우에는 신뢰되지 않은 사용자이거나 신뢰되지 않은 웹사이트인 경우에 주의가 필요하다.
3) Anti-Virus 제품 및 개인 방화벽을 사용한다.
4) 네트워크 관리자는 네트워크 보안 제품의 사용을 고려한다.
5) 네트워크 관리자는 메일 서버에서 오피스 파일이 첨부된 이메일(E-Mail)을 필터링(Filtering)하는 것을 고려할 수도 있다.

III. ASEC 컬럼

(1) 악성코드 분석가 입장에서 본 PE 구조

1. 악성코드와 PE(Portable Executable)파일의 조우

PE 파일이라 부르는 형식은 플랫폼에 관계없이 Win32 운영체제 시스템이면 어디든 실행 가능한 프로그램을 뜻한다. PE 파일(*.EXE, *.DLL)의 실행을 위해서는 운영체제에 실행 파일의 정보를 제공할 필요가 있는데, 예를 들면 실행 파일의 기계어 코드 위치, 아이콘 및 그림 파일 등의 위치, 해당 파일이 실행될 수 있는 플랫폼의 종류, 운영체제가 파일을 실행 시킬 때 첫 시작 코드의 위치 등 수많은 정보를 제공하여야 한다. 이와 같은 다양한 정보들이 저장된 곳이 PE 파일의 처음에 위치한 PE 헤더 구조체이다.

악성 코드를 분석하여 보면 PE 헤더 구조체에 흥미로운 정보들이 많이 존재하고 있음을 알 수 있다. 악성 코드의 크기를 줄이기 위해 헤더 정보를 속이거나, 바이러스에 감염되어 원본 파일의 헤더가 변경 되기도 한다. 또한 특정 악성코드만의 고유한 정보가 숨어 있기도 하며, 일반 정상파일에서는 있을 수 없는 값들이 들어 있기도 하다.

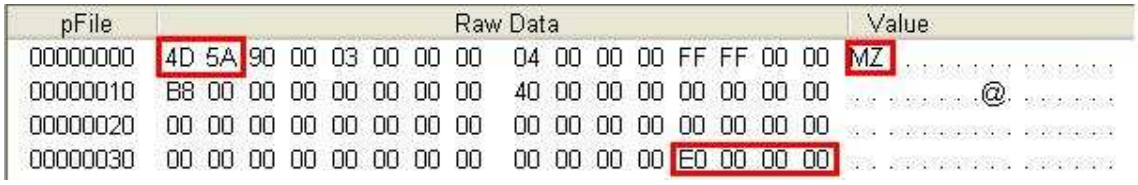
2. 현재의 악성코드가 DOS 시절의 헤더를 이용한다?

PE 파일은 IMAGE_DOS_HEADER 구조체로 시작한다. 이는 DOS 시절에 사용되던 실행 파일의 헤더로서 실행 파일이 DOS에서 실행 되었을 때 DOS 운영체제에 알려줘야 할 정보들이 담겨 있다. 다음은 IMAGE_DOS_HEADER 구조체이다.

```
typedef struct _IMAGE_DOS_HEADER        // DOS .EXE header
{
    WORD e_magic;           // Magic number
    WORD e_cblp;            // Bytes on last page of file
    WORD e_cp;              // Pages in file
    WORD e_crlc;            // Relocations
    WORD e_cparhdr;         // Size of header in paragraphs
    WORD e_minalloc;        // Minimum extra paragraphs needed
    WORD e_maxalloc;        // Maximum extra paragraphs needed
    WORD e_ss;              // Initial (relative) SS value
    WORD e_sp;              // Initial SP value
    WORD e_csum;            // Checksum
    WORD e_ip;              // Initial IP value
```

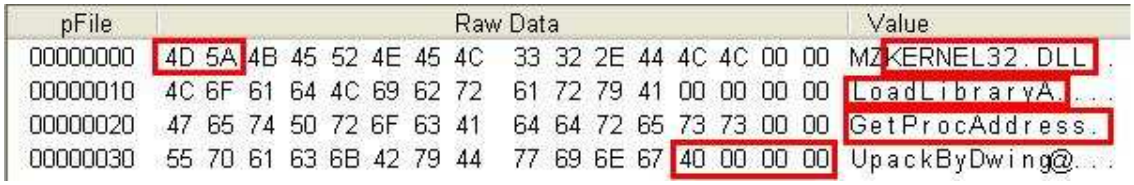
```
    WORD e_cs;              // Initial (relative) CS value
    WORD e_lfalc;           // File address of relocation table
    WORD e_ovno;            // Overlay number
    WORD e_oemid;           // OEM identifier (for e_oeminfo)
    WORD e_oeminfo;         // OEM information; e_oemid specific
    WORd e_res2[10];         // Reserved words
    LONG e_lfanew;           // File address of new exe header
} IMAGE_DOS_HEADER, *PIMAGE_DOS_HEADER;
```

그러나, 이것은 DOS 운영체제를 위해 제공하는 정보들이기 때문에 현재 Windows 환경에서는 필요치 않다. 대다수의 필드들은 무시되며 e_magic 및 e_lfanew 필드만이 유용한 정보가 된다. e_magic 필드는 IMAGE_DOS_HEADER의 시작을 나타내는 것으로 항상 'MZ'라는 문자열로 시작을 한다. 그리고 e_lfanew 필드는 IMAGE_DOS_HEADER 다음에 나오는 헤더 파일의 오프셋 값을 가지고 있다. 그 외의 필드들은 Windows에서 파일을 실행시켰을 때 이용되지 않는다. 다음 [그림 3-1]은 일반적인 실행 파일의 IMAGE_DOS_HEADER의 값이다.



[그림 3-1] IMAGE_DOS_HEADER 정보

이 위의 값들은 만일 DOS Mode 실행 시 Dos Stub 실행을 목적으로 지정된 값들로 이 헤더 뒤에 따르는 문자열 "This program cannot be run in DOS mode"을 출력해 주기 위해 지정된 값들이다. Dos가 아닌 Windows 모드에서 실행 시켰을 경우에는 이용되지 않는다. 그러나 몇몇 악성코드를 분석하다 보면 실제 사용되지 않는 필드 값들이 악성코드에 의해 사용되고 있는 흥미로운 사실을 발견할 수 있다. 다음 [그림 3-2]는 악성코드에서 많이 사용되는 실행압축 방식의 하나인 Upack의 IMAGE_DOS_HEADER 부분이다.



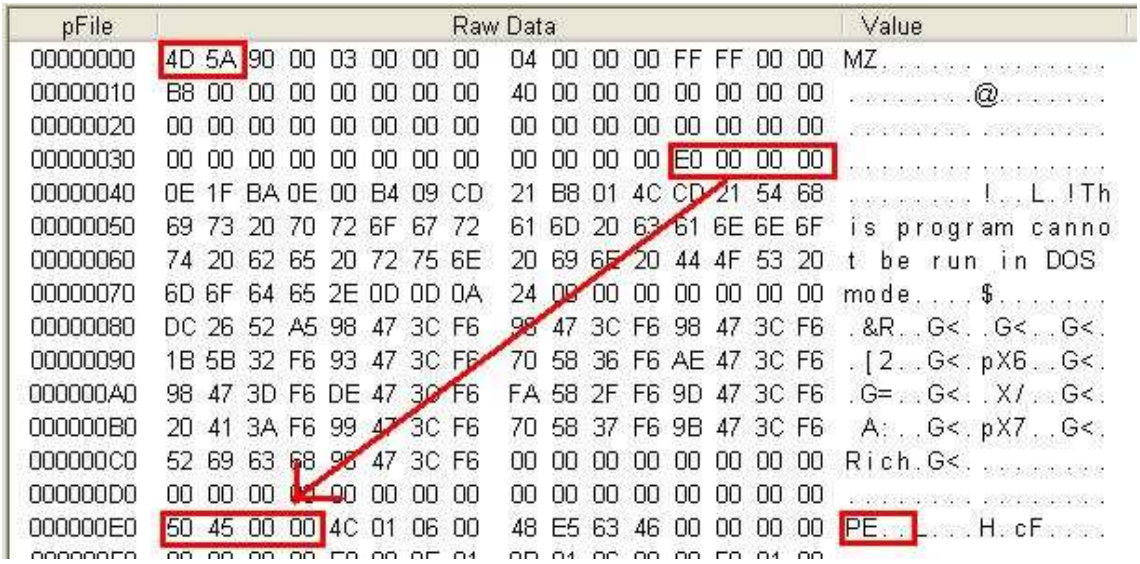
[그림 3-2] Upack의 IMAGE_DOS_HEADER 정보

앞서 설명한 IMAGE_DOS_HEADER 와는 확연한 차이를 보인다. e_magic 및 e_lfanew 필드 이외의 값들이 'KERNEL32.DLL', 'LoadLibraryA', 'GetProcAddress' 등의 문자열로 채

워진 것을 확인 할 수 있다. 이들 문자열들은 동적링크를 위한 함수 호출에 필요한 문자열들로 실행압축인 Upack에서 필요한 라이브러리 함수(DLL)들을 가져다 쓰는 루틴을 위해 존재한다. 여기서는 범용 실행압축 모듈인 Upack을 통해 실제 이용되지 않는 필드들을 활용하는 사례를 다루었지만, 악성코드들 역시 필요한 정보들의 보관을 위해 IMAGE_DOS_HEADER의 빈 공간을 적절히 이용하고 있다.

3. e_lfanew 필드가 IMAGE_DOS_HEADER 범위 안을 가리키고 있다?

IMAGE_DOS_HEADER의 e_lfanew 필드는 다음에 올 헤더 위치의 파일 오프셋을 가리키고 있다. 즉, 실제 PE 파일의 시작이라고 할 수 있는 IMAGE_NT_HEADER의 시작 오프셋 값을 가지고 있다. 다음 [그림 3-3]은 일반적인 실행 파일의 e_lfanew 필드의 값인 0x0000000E 위치의 값을 보여준다.

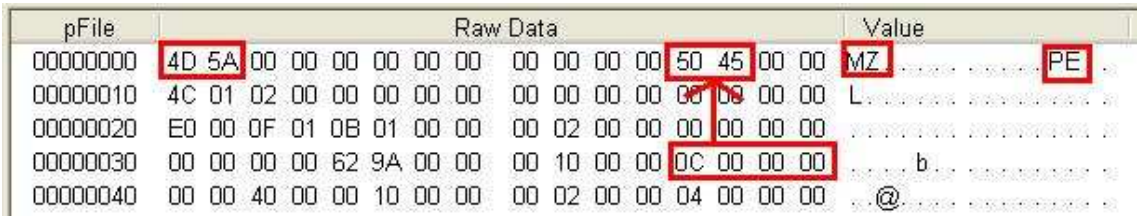


[그림 3--3] IMAGE_DOS_HEADER의 e_lfanew 정보

위 그림과 같이 일반적인 실행 파일은 다음에 올 헤더 위치의 파일 오프셋을 가리키고 있다. 0x000000E0 위치에 있는 헤더는 IMAGE_NT_HEADER로써 다음과 같은 구조체로 정의가 되어 있다.

```
typedef struct _IMAGE_NT_HEADERS
{
    DWORD Signature;
    IMAGE_FILE_HEADER FileHeader;
    IMAGE_OPTIONAL_HEADER32 OptionalHeader;
} IMAGE_NT_HEADER32, *PIMAGE_NT_HEADER32;
```

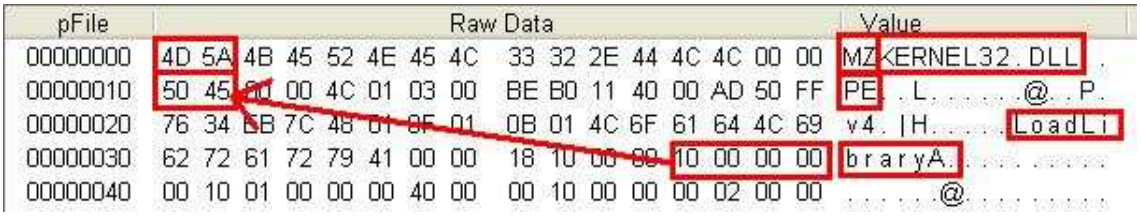
첫 필드인 Signature는 PE 헤더의 시작을 알리는 값으로 'PE'라는 문자열이 들어가게 된다. 그렇기 때문에 IMAGE_DOS_HEADER의 e_lfanew 필드가 가리키는 위치를 가보면 'PE₩0₩0'과 같은 문자열이 나타나게 된다. IMAGE_FILE_HEADER와 IMAGE_OPTIONAL_HEADER32는 PE 구조체의 세부적인 값들로 지정된 구조체이다. 그럼 e_lfanew 필드는 언제나 자신보다 뒤에 있는 영역을 가리키고 있는 것일까? 일반적인 실행파일의 경우 그렇다고 할 수 있다. 그러나 몇몇 악성코드들에서는 다음 [그림 3-4]와 같은 모습이 보여지기도 한다.



[그림 3-4] 악성코드에서의 특이한 e_lfanew 정보의 예

일반적인 실행파일과 다르게 IMAGE_DOS_HEADER의 e_lfanew 필드가 자신보다 앞의 영역을 가리키고 있다. 이와 같이 다소 일반적이지 않은 형태로 구조체가 정의 되더라도 각 필드에 맞는 값들이 채워진다면 운영체제가 파일을 읽어 들여서 실행하는 것에는 문제가 되지 않는다. 특히 IMAGE_DOS_HEADER의 첫 필드와 마지막 필드를 제외하고는 현재 이용되지 않기 때문에 전혀 문제가 되지 않는다. 단, IMAGE_NT_HEADER 구조체의 시작이 앞으로 이동하게 되면 e_lfanew 필드 값이 PE 헤더 값과 겹치게 되므로, e_lfanew 필드 값과 매칭되는 PE 헤더의 값(BaseOfData)에 문제가 없어야만 한다.

다음 [그림 3-5]를 살펴보면 IMAGE_DOS_HEADER의 e_lfanew 필드는 파일 오프셋으로 0x00000010을 가리키고 있다. 그리고 IMAGE_NT_HEADER의 앞부분에 위치한 이용되지 않는 IMAGE_DOS_HEADER의 영역에 'KERNEL32.DLL' 문자열을 숨겨 두었다. IMAGE_NT_HEADER를 보면 PE 헤더의 시작 뒤를 보면 파일 오프셋 0x0000002A 부분부터 'LoadLibraryA' 문자열이 있는 것을 확인할 수 있다.



[그림 5] 악성코드가 삽입한 IMAGE_DOS_HEADER 내의 문자열 정보

해당 영역은 IMAGE_NT_HEADER에 멤버로 등록된 구조체인

IMAGE_OPTIONAL_HEADER의 영역으로 각 대칭되는 필드는 다음 그림 [2-6]과 같다.

pFile	Data	Description
00000028	010B	Magic
0000002A	4C	Major Linker Version
0000002B	6F	Minor Linker Version
0000002C	694C6461	Size of Code
00000030	72617262	Size of Initialized Data
00000034	00004179	Size of Uninitialized Data
00000038	00001018	Address of Entry Point
0000003C	00000010	Base of Code
00000040	00011000	Base of Data
00000044	00400000	Image Base

[그림 2-6] IMAGE_OPTIONAL_HEADER 정보

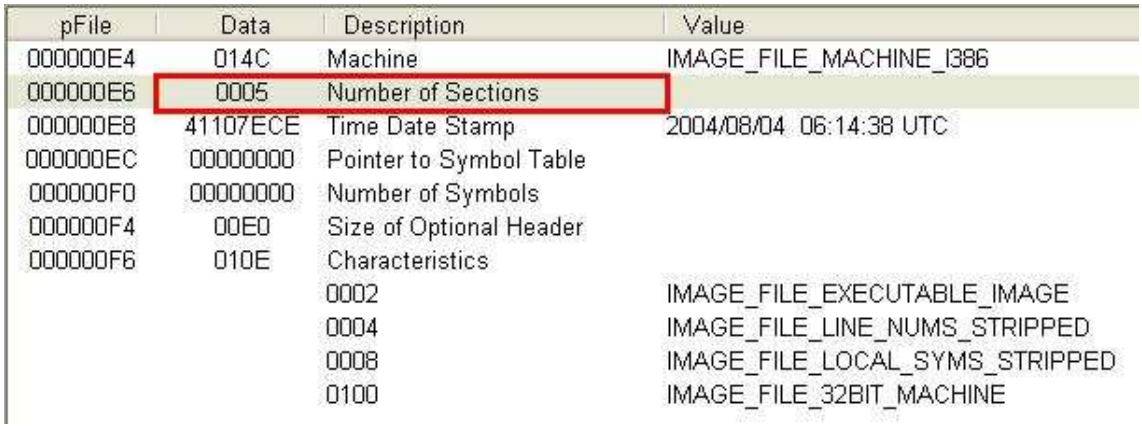
Major Linker Version, Minor Linker Version, SizeOfCode, SizeOfInitializedData, SizeOfUninitializedData의 값이 된다. Major 및 Minor Linker Version 필드는 실행파일을 만든 링커의 버전을 담고 있으며, SizeOfCode 필드는 모든 코드 섹션들의 사이즈를 합한 크기이다. SizeOfInitializedData 필드는 코드 섹션을 제외한 초기화된 데이터 섹션의 전체 크기를 나타내며, SizeOfUninitializedData 필드는 초기화되지 않은 데이터 섹션의 바이트 수를 나타낸다. 일반적인 실행파일의 경우 이러한 값들은 거의 쓰이지 않는다.

4. 다른 컴퓨터의 정상적인 실행 파일과 내 컴퓨터의 정상적인 실행 파일이 다르다?

일반적으로 같은 플랫폼에서의 같은 실행 파일의 경우 직접 링커를 통해 실행 파일을 만들지 않는 이상(즉, 일반적으로 제공되는 어플리케이션을 설치했을 경우에는) 같은 속성을 가지게 된다. 그러나 이 실행 파일 헤더의 특정 몇몇 부분이 다를 경우에는 실행 파일이 감염되었음을 의심해 볼 수 있다. [그림 3-7]과 [그림 3-8]에서 진단명 Tufic.C에 감염 전후의 explorer.exe 파일의 차이를 확인해볼 수 있다..

pFile	Data	Description	Value
000000E4	014C	Machine	IMAGE_FILE_MACHINE_I386
000000E6	0004	Number of Sections	
000000E8	41107ECE	Time Date Stamp	2004/08/04 06:14:38 UTC
000000EC	00000000	Pointer to Symbol Table	
000000F0	00000000	Number of Symbols	
000000F4	00E0	Size of Optional Header	
000000F6	010E	Characteristics	
		0002	IMAGE_FILE_EXECUTABLE_IMAGE
		0004	IMAGE_FILE_LINE_NUMS_STRIPPED
		0008	IMAGE_FILE_LOCAL_SYMS_STRIPPED
		0100	IMAGE_FILE_32BIT_MACHINE

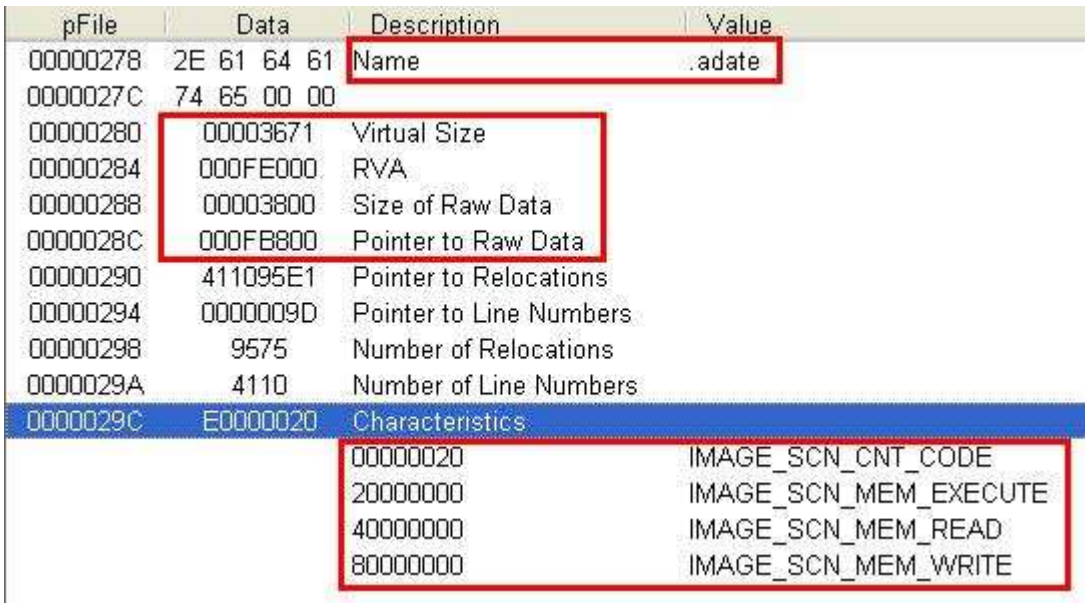
[그림 7] Tufic.C 감염 전 Number of Sections 정보



pFile	Data	Description	Value
000000E4	014C	Machine	IMAGE_FILE_MACHINE_I386
000000E6	0005	Number of Sections	
000000E8	41107ECE	Time Date Stamp	2004/08/04 06:14:38 UTC
000000EC	00000000	Pointer to Symbol Table	
000000F0	00000000	Number of Symbols	
000000F4	00E0	Size of Optional Header	
000000F6	010E	Characteristics	
		0002	IMAGE_FILE_EXECUTABLE_IMAGE
		0004	IMAGE_FILE_LINE_NUMS_STRIPPED
		0008	IMAGE_FILE_LOCAL_SYMS_STRIPPED
		0100	IMAGE_FILE_32BIT_MACHINE

[그림 8] Tufic.C 감염 후 Number of Sections 정보

[그림 2-7]은 Tufic.C 감염되기 이전의 IMAGE_NT_HEADER에 속한 IMAGE_FILE_HEADER의 값이며, [그림 2-8]은 감염된 이후의 모습이다. 박스안의 NumberOfSections 필드를 살펴보면 감염된 이후에 값이 1만큼 증가한 것을 확인 할 수 있다. Tufic.C는 원본 파일의 맨 뒤에 바이러스를 첨가 시키는 Appending 바이러스의 한 유형으로 바이러스를 추가할 부분을 만들기 위해 섹션의 수를 하나 증가시켜 놓은 것이다.

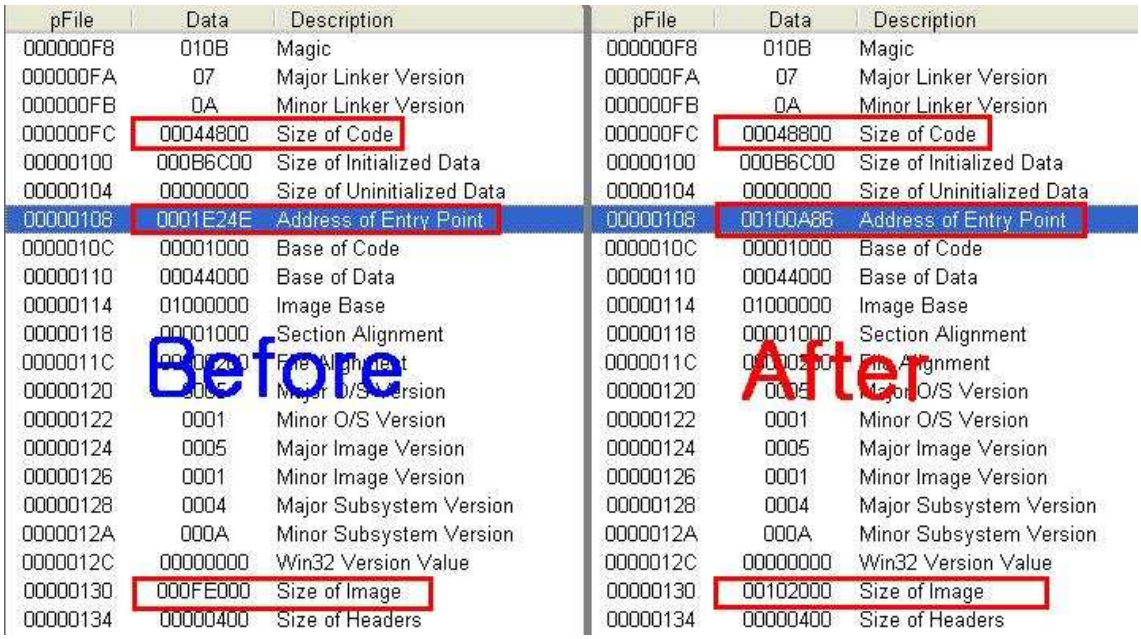


pFile	Data	Description	Value
00000278	2E 61 64 61	Name	.adate
0000027C	74 65 00 00		
00000280	00003671	Virtual Size	
00000284	000FE000	RVA	
00000288	00003800	Size of Raw Data	
0000028C	000FB800	Pointer to Raw Data	
00000290	411095E1	Pointer to Relocations	
00000294	0000009D	Pointer to Line Numbers	
00000298	9575	Number of Relocations	
0000029A	4110	Number of Line Numbers	
0000029C	E0000020	Characteristics	
		00000020	IMAGE_SCN_CNT_CODE
		20000000	IMAGE_SCN_MEM_EXECUTE
		40000000	IMAGE_SCN_MEM_READ
		80000000	IMAGE_SCN_MEM_WRITE

[그림 2-9] Tufic.C 감염후 추가된 섹션 정보

[그림 2-9]는 추가된 섹션 헤더의 모습이다. 여기에서 상당히 많은 악성코드 정보를 얻을 수 있으며, 이러한 정보는 악성코드 분석에 큰 도움을 준다. 우선 섹션의 이름은 .adate임을 알 수 있다. Tufic.C와 같이 바이러스 감염 시 섹션 이름을 특정 문자열로 주게되면, 바이러

스 감염여부에 대한 기초적인 판단 정보가 되기도 한다. VirtualSize는 감염 파일이 운영체제에 의해 로드 되었을 때, 이 섹션의 이미지상의 크기를 나타내어 준다. 이것은 바이러스 코드의 길이를 짐작 할 수 있다. RVA는 실행 파일이 운영체제에 의해 로드 되었을 때 메모리상의 위치 정보이며, 감염 파일을 분석할 때 바이러스 코드를 확인 할 수 있다. SizeOfRawData는 파일상에서 섹션의 크기에 대해 알 수 있으며, 치료할 때 사용 된다. PointerToRawData는 파일상의 섹션 위치를 알 수 있으며 이것 또한 치료할 때 사용 된다. Characteristics은 해당 섹션의 속성을 나타내는 플래그의 집합이다. 바이러스가 생성한 섹션의 플래그는 IMAGE_SCN_CNT_CODE, _EXECUTE, _READ, _WRITE 등의 속성이 지정되어 있으며 각각은 '코드를 포함하고 있다', '실행 가능한 섹션이다', '읽기 가능 섹션이다', '쓰기 가능 섹션이다'는 것을 의미한다. [그림 3-10]에서와 같이 섹션 정보 이외에 수정되는 것이 더 있는지 확인을 해 보면 중간 부분에서와 같이 몇 개의 필드가 수정된 것을 확인할 수 있다.



pFile	Data	Description
000000F8	010B	Magic
000000FA	07	Major Linker Version
000000FB	0A	Minor Linker Version
000000FC	00044800	Size of Code
00000100	000B6C00	Size of Initialized Data
00000104	00000000	Size of Uninitialized Data
00000108	0001E24E	Address of Entry Point
0000010C	00001000	Base of Code
00000110	00044000	Base of Data
00000114	01000000	Image Base
00000118	00001000	Section Alignment
0000011C	[illegible]	File Alignment
00000120	[illegible]	Major O/S Version
00000122	0001	Minor O/S Version
00000124	0005	Major Image Version
00000126	0001	Minor Image Version
00000128	0004	Major Subsystem Version
0000012A	000A	Minor Subsystem Version
0000012C	00000000	Win32 Version Value
00000130	000FE000	Size of Image
00000134	00000400	Size of Headers

pFile	Data	Description
000000F8	010B	Magic
000000FA	07	Major Linker Version
000000FB	0A	Minor Linker Version
000000FC	00048800	Size of Code
00000100	000B6C00	Size of Initialized Data
00000104	00000000	Size of Uninitialized Data
00000108	00100A86	Address of Entry Point
0000010C	00001000	Base of Code
00000110	00044000	Base of Data
00000114	01000000	Image Base
00000118	00001000	Section Alignment
0000011C	[illegible]	File Alignment
00000120	[illegible]	Major O/S Version
00000122	0001	Minor O/S Version
00000124	0005	Major Image Version
00000126	0001	Minor Image Version
00000128	0004	Major Subsystem Version
0000012A	000A	Minor Subsystem Version
0000012C	00000000	Win32 Version Value
00000130	00102000	Size of Image
00000134	00000400	Size of Headers

[그림 3-10] Tufic.C 감염 전/후의 IMAGE_OPTIONAL_HEADER 정보

IMAGE_NT_HEADER에 속해 있는 IMAGE_OPTIONAL_HEADER32 구조체의 필드 값들이다. 왼쪽이 감염 이전의 explorer.exe 파일이며, 오른쪽이 감염 이후의 explorer.exe 파일이다. 감염 전/후를 비교해 보면, 세 부분이 바뀐 것을 확인 할 수 있는데, 첫 번째 SizeOfCode는 앞서 설명과 같이 모든 코드 섹션의 사이즈를 합한 크기이며 그 크기가 증가한 것을 확인 할 수 있다. 또한 맨 마지막의 SizeOfImage는 운영체제가 이 실행 파일을 로드할 때 확보/예약해야 할 메모리상의 크기를 가리킨다. 이 또한 SizeOfCode가 증가한 크기만큼 증가한 것을 확인 할 수 있다.

중요한 것은 AddressOfEntryPoint 인데 이 부분은 운영체제가 실행 파일을 로드 했을 때, 이 실행 파일의 코드 시작점을 나타낸다. Tufic.C에 감염 되었을 경우에는 이와 같이 코드 진입점이 변경되며 변경된 코드 진입점은 바이러스의 시작 주소를 가리키게 된다. 이것은 바이러스 분석에 결정적 자료가 된다.(물론 이처럼 AddressOfEntryPoint 필드를 변경하지 않고, 실제 시작 코드를 변경하는 바이러스들도 존재한다.) 변경된 원본 AddressOfEntryPoint는 바이러스가 나중에 원본 파일을 정상 실행 시키기 위해서 임의의 위치에 백업을 해 둔다.

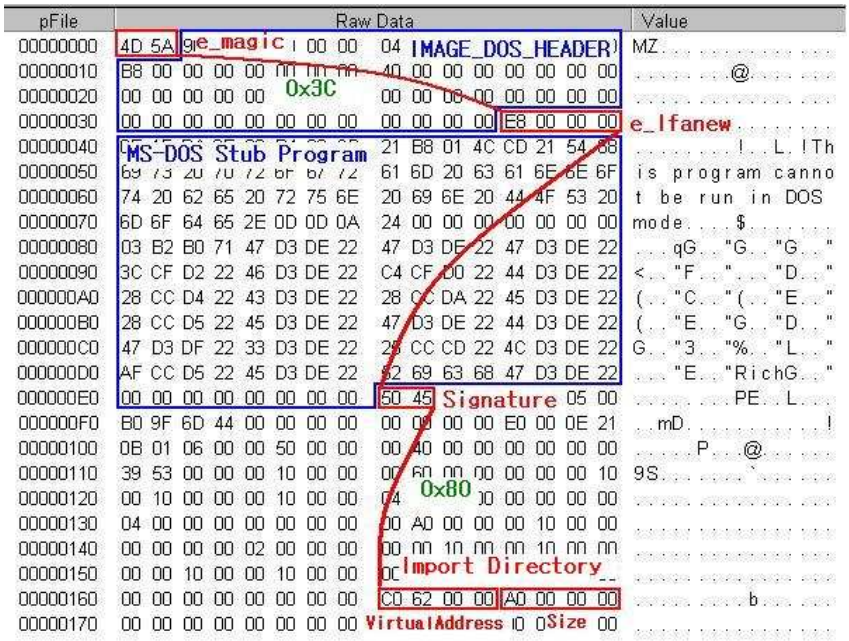
5. Import Address Table 정보를 이용한 악성코드 분석

DLL(Dynamic Link Library)은 서로 다른 프로그램이 실행된 후 공통적으로 사용하는 함수들을 하나로 묶어두고 이를 필요로 할 때 동적으로 링크하여 사용하는 공유 라이브러리 파일이다. 각각의 프로그램에서 하나의 함수를 공통적으로 사용하게 되므로 메모리가 절약되고, 주 프로그램과 함께 컴파일 되는 정적 링크에 비해 상대적으로 작은 사이즈를 가지는 잇점이 있다. 최대한 간략하게 만들어야 하는 악성코드 역시 필요한 함수를 동적 링크 하는 것이 일반적이므로, 함께 링크되는 DLL 함수 정보를 살펴보면 악성코드가 의도하는 목적을 유추할 수 있다. PE 구조에서 해당 정보를 보관하고 있는 Import Address Table을 찾아가 보기로 한다.

1) PE 파일의 시작(e_magic)에서 0x3C 만큼 이동하면 IMAGE_NT_HEADERS 시작 오프셋 값(e_lfanew)을 찾을 수 있다

2) IMAGE_NT_HEADERS 시작 오프셋(Signature)에서 0x80 만큼 이동한 지점이 Import Directory RVA 구조체 정보이다. 구조체 엔트리에 대한 의미는 WinNT.H 파일에 다음과 같이 정의되어 있다.

```
typedef struct _IMAGE_DATA_DIRECTORY {
        DWORD VirtualAddress;
        DWORD Size;
} IMAGE_DATA_DIRECTORY, *PIMAGE_DATA_DIRECTORY;
```



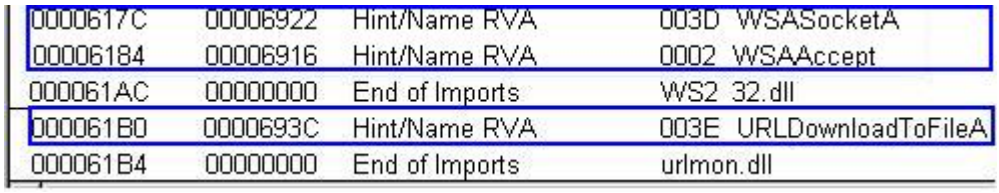
[그림 3-11] Import Directory RVA 정보

3) Import Directory RVA에 대한 FileOffset으로 이동하면 동적으로 링크하는 DLL 파일들의 구조체(IMAGE_IMPORT_DESCRIPTOR)정보를 확인할 수 있다. 다음은 Win-Trojan/Backdoor.40960.B 에서 동적으로 링크하는 DLL 및 그에 따른 함수(Import Address Table) 정보를 일부 발췌한 것이며, 이를 통해 해당 악성코드는 URL 접속 및 정보 유출, 불특정 파일 다운로드 등의 증상을 유추해 볼 수 있겠다.

- urlmon.dll: Internet Explorer의 구성요소로, 웹사이트에서 반환된 URL과 정보를 처리한다
- WS2_32.dll: Windows Socket API, WSAStartup 함수를 통해 DLL을 로딩하여 사용한다

00006310	000064D0	Import Name Table RVA	
00006314	00000000	Time Date Stamp	
00006318	00000000	Forwarder Chain	
0000631C	00006930	Name RVA	WS2_32.dll
00006320	00006170	Import Address Table RVA	
00006324	00006510	Import Name Table RVA	
00006328	00000000	Time Date Stamp	
0000632C	00000000	Forwarder Chain	
00006330	00006952	Name RVA	urlmon.dll
00006334	000061B0	Import Address Table RVA	

[그림 3-12] IMAGE_IMPORT_DESCRIPTOR 정보 (Win-Trojan/Backdoor.40960.B)



0000617C	00006922	Hint/Name RVA	003D WSASocketA
00006184	00006916	Hint/Name RVA	0002 WSAAccept
000061AC	00000000	End of Imports	WS2_32.dll
000061B0	0000693C	Hint/Name RVA	003E URLDownloadToFileA
000061B4	00000000	End of Imports	urlmon.dll

[그림 3-13] Import Address Table 정보 (Win-Trojan/Backdoor.40960.B)

6. Import Address Table 정보를 은닉한 악성코드 분석

레지스트리 및 파일 I/O, 소켓연결, 프로세스 관련 함수 호출은 악성코드들이 즐겨 사용하며, 이를 통하여 안티바이러스 프로그램 및 분석가들도 악성 여부를 판단하는 기초 자료로 사용한다.

이러한 특성을 인지한 악성코드 제작자들은 치료백신 업데이트 및 분석 지연을 목적으로 DLL 정보 및 호출함수 정보를 은닉하게 되는데 실행압축(Packer), 암호화(Encrypt)등을 사용하는 것이 일반적이며, 일부 악성코드는 PE 구조체의 Import Address Table 정보를 메인 루틴 에서 생성하기도 한다.

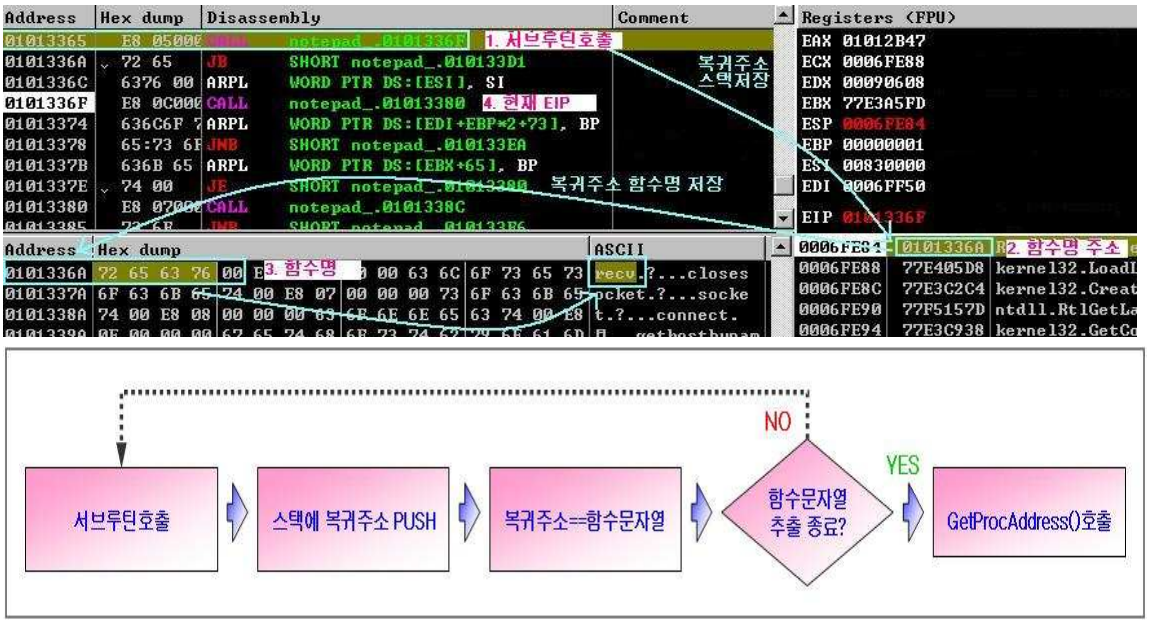
1) GetProcAddress() 함수주소 획득

호출 함수들의 주소정보 획득을 위해서 DLL 핸들, 함수명을 인자값으로 GetProcAddress()를 호출하며, GetProcAddress() 함수에 대한 주소정보는 메인루틴 상의 kernel32.dll 에서 획득한다. 다음은 Win32/MaDang 에서 GetProcAddress() 함수 주소를 획득하는 과정을 보여준다.

[그림 3-14] GetProcAddress 함수에 대한 주소획득 과정 (Win32/MaDang)

2) 함수명 추출을 위한 서브루틴 호출

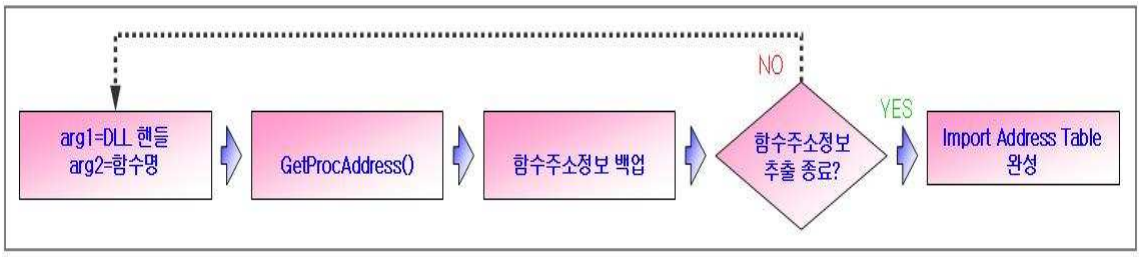
LoadLibrary()를 통해 DLL핸들이 확보되면, GetProcAddress() 인자값으로 사용할 함수명을 추출해야 하는데, Win32/MaDang 에서는 다소 변칙적인 함수호출 과정을 이용한다. 일반적인 함수 호출은 서브루틴이 호출되면 복귀주소(Return Address)가 스택에 쌓이고(PUSH) 서브루틴 종료시 복귀주소를 스택에서 꺼내어(POP) 메인루틴으로 복귀하는 과정을 밟는다. 다음은 Win32/MaDang 에서 위와 같은 함수호출 과정으로 스택에 쌓인 값이 복귀주소를 가리키지 않고 함수명을 가리킬 수 있음을 보여준다.



[그림 3-15] 함수명 추출을 위한 서브루틴 호출 과정 (Win32/MaDang)

3) 호출 함수들의 주소정보 획득

다소 변칙적인 함수호출 과정을 통해 필요한 함수명이 스택에 저장되었고 DLL핸들 또한 확보되었다. 최종적으로 GetProcAddress() 함수 호출을 통해 메인루틴에서 사용될 함수들의 주소정보를 모두 획득하면 Import Address Table 이 완성된다.



[그림 3-16] 호출 함수들의 주소정보 획득 과정 (Win32/MaDang)

7. PE(Portable Executable) 구조체와 악성 코드 분석가의 조우

악성코드를 분석하다 보면 이 이외에도 많은 흥미로운 것들을 발견할 수 있다. 그러나 앞에서 언급을 했듯이 모든 악성코드들이 이러한 특성들을 가지고 있는 것은 아니며, 정상 파일은 이러한 특성을 가지지 않는 것은 아니다. 다만 악성코드들에서 발생 빈도가 더 높을 뿐이다. 이러한 정보들은 악성코드의 악의적인 동작들과는 관계가 없는 것들은 많지만, 악성코드를 분석하고 악성/정상을 판단하는 것에 결정적인 힌트가 되어 주기도 한다. 또한 각 악성코드만의 PE 헤더 특성을 악성코드 진단에 이용할 수 있으며 바이러스와 같은 경우에는 정상 파일을 감염시켜 PE 헤더의 내용을 바꾸는 동작을 수행할 가능성이 높기 때문에 치료함수를 제작하기 위해서 이러한 PE 헤더의 바뀐 부분에 대한 파악이 중요하다. 그러므로 PE 구조체와 악성코드 분석은 떼어낼래야 뗄 수 없는 관계이다